

Departamento de Engenharia Electrotécnica e de Computadores da
Universidade de Coimbra

12-13 November 2015



**ENCONTRO PORTUGUÊS
DE
COMPUTAÇÃO GRÁFICA
E
INTERAÇÃO**



Instituto de Sistemas e Robótica

Grupo Português de Computação Gráfica

Atas do 22º Encontro Português de Computação Gráfica e Interação

Departamento de Engenharia Eletrotécnica e de Computadores
Universidade de Coimbra, Polo II
3030-290 Coimbra, Portugal
12 e 13 de novembro de 2015

Edição: Paulo Menezes

Publicado por:
Instituto de Sistemas e Robótica
Universidade de Coimbra, Polo II
3030-290 Coimbra, Portugal

PDF ISBN: 978-989-20-6383-6

Capa:
Interior do Criptopórtico do Forum de Aeminium - Museu Machado de Castro - Coimbra
Fotografia tirada por Luís Almeida para SciTecIN'15.



Preface

These proceedings are a collection of selected papers presented at the twenty second Portuguese Meeting on Computer Graphics and Interaction, jointly organised with the Portuguese Conference on the Sciences of Video Games, under the name Sciences and Technologies of Interaction (SciTecIN'15). The organisation of this event was the responsibility of the Instituto de Sistemas e Robótica (ISR-UC) on behalf of Grupo Português de Computação Gráfica. This conference aimed at continuing a quality tradition that had been maintained along the twenty one previous editions.

With the objectives of bringing together researchers and professionals of the fields related with computer graphics and interactive systems, works were accepted in both Portuguese and English languages, as happened with the previous editions. We believe that this may broaden the spectrum of the audience of the presented conference works. Nevertheless, it was the authors choice of either using English or Portuguese language in their articles. Following the same principles, these proceedings were also edited in order of keeping their bilingual characteristic.

The contributed papers were reviewed and evaluated by the Technical Committee members, resulting in a selection of papers for oral or poster presentation. The oral sessions were divided by the following thematic areas: Interaction Design; Virtual, Mixed and Augmented Reality; Interactive Computer Graphics; and Computer Vision & 3D. The “Poster Session”, happening simultaneously with a “Demonstrators Session”, revealed to be an excellent way of establishing close connections with the interested participants. This was due to the possibility of directly posing and answering questions without the pressure imposed by the limited time of typical oral sessions. Beyond these sessions, the conference program counted with three invited speakers: Jefferson Valadares, Julien Pettré and Miguel Sales Dias. The conference was closed with the delivery of the prize “Professor José Luís Encarnação”, followed by the delivery of the certificates of best conference article and best demonstrator.

The organisers would like to thank the authors, the invited speakers, the members of the Technical Committee, the members of the Organising Committee, ISR, and DEEC-UC for the contributions, help and support. A special thank to the Grupo Português de Computação Gráfica, whose engagement and efforts has made possible this series of conferences to happen along the years.

Kind regards,

José Carlos Teixeira
Paulo Dias
Paulo Menezes



Prefácio

Estas atas contêm a coleção dos artigos que foram selecionados e apresentados no vigésimo segundo Encontro Português de Computação Gráfica, o qual foi organizado em conjunto com a conferência Video Jogos, sob o nome Ciências e Tecnologias da Interação (SciTecIN'15). A organização deste evento foi levada a cabo pelo Instituto de Sistemas e Robótica (ISR-UC), em nome do Grupo Português de Computação Gráfica, tendo desde o início sido definido o objectivo de manter a tradição de qualidade patente nas edições anteriores. Tendo como objectivo a reunião de investigadores e profissionais das áreas relacionadas com a computação gráfica e os sistemas interativos, foram aceites trabalhos tanto em língua Portuguesa como Inglesa, tal como aconteceu com as edições anteriores. Acreditamos que esta característica bilingue permite alargar a audiência dos trabalhos apresentados. No entanto, a escolha da língua usada em cada um dos artigos foi da responsabilidade dos próprios autores. Assim, e seguindo estes princípios, estas atas foram editadas de forma a manter esta característica bilingue dos trabalhos incluídos.

Os artigos enviados pelos autores foram revistos e avaliados pelos membros da Comissão Científica, resultando na seleção de artigos aqui incluída, e que foram apresentados oralmente ou sob a forma de poster. As sessões orais dividiram-se pelas seguintes áreas: Design de Interação; Realidade Virtual, Mista e Aumentada; Computação Gráfica Interativa; Visão Computacional e 3D. A sessão de “Posters” foi realizada em simultâneo com uma sessão de “Demonstradores”, o que se revelou ser uma excelente forma de comunicar com os participantes interessados. Isto deveu-se à possibilidade dos interessados poderem colocar questões diretamente aos autores e obter as respetivas respostas, sem a pressão imposta normalmente pelas limitações temporais das sessões orais típicas. Para lá destas sessões, a conferência contou ainda com três oradores convidados; Jefferson Valadares, Julien Pettré e Miguel Sales Dias.

A conferência foi encerrada com a entrega do prémio “Professor José Luís Encarnação”, a que se seguiu a entrega dos certificados de melhor artigo e melhor demonstrador apresentados nesta conferência.

Os organizadores gostariam de agradecer aos autores, aos oradores convidados, aos membros da Comissão Científica, aos membros da Comissão Organizadora, ao ISR-UC e ao DEEC-UC pelas contribuições, ajuda e suporte que permitiram a realização deste evento. Um agradecimento especial ao Grupo Português de Computação Gráfica, cuja dedicação e esforço tem permitido que a realização destas conferências se tenha mantido ao longo dos anos.

Cordiais Saudações,

José Carlos Teixeira
Paulo Dias
Paulo Menezes



Technical Committee / Comissão Científica

Chairs/Presidentes:

Paulo Dias	IEETA / Universidade de Aveiro
Paulo Menezes	ISR-UC / Universidade de Coimbra

Members/Membros:

A Augusto Sousa	INESC TEC / Faculdade de Engenharia da Universidade do Porto
Abel Gomes	Universidade da Beira Interior
Adérito Fernandes-Marcos	Universidade Aberta
Alexandrino Gonçalves	Instituto Politécnico de Leiria
Ana Paula Cláudio	BioISI / Faculdade de Ciências da Universidade de Lisboa
Antonio Coelho	Faculdade de Engenharia da Universidade do Porto
Antonio Costa	Instituto Superior de Engenharia do Porto
Antonio Marin-Hernandez	Universidad Veracruzana, Xalapa, Mexico
António Ramires	Universidade do Minho
Beatriz Sousa Santos	Universidade de Aveiro
Bruno Patrão	ISR-UC / Universidade de Coimbra
Daniel Gonçalves	INESC-ID
Emanuel Peres	Universidade de Trás-os-Montes e Alto Douro
Francisco Pereira	Universidade de Trás-os-Montes e Alto Douro
Frutuoso Silva	Universidade da Beira Interior
Helder Araujo	ISR-UC / Universidade de Coimbra
João Paulo Moura	Universidade de Trás-os-Montes e Alto Douro
João Tavares	Faculdade de Engenharia da Universidade do Porto
Joaquim Armando Jorge	INESC-ID
Joaquim Madeira	DETI / IEETA, Universidade de Aveiro
Jorge Dias	ISR-UC / Universidade de Coimbra
Jorge S. Marques	ISR - IST, Universidade de Lisboa
José Campos	Universidade do Minho
José Carlos Teixeira	IT/DMat-FCT- Universidade de Coimbra
Jose-Joel Gonzalez-Barbosa	CICATA, Instituto Politecnico Nacional, Mexico

Juan Carlos García Sánchez	Universitat Jaume I, Castellón, Spain
Julien Pettré	INRIA, Rennes, France
Luis Almeida	ISR-UC / Instituto Politécnico de Tomar
Luís Gonzaga Magalhães	Universidade do Minho
Luis Marcelino	Instituto Politécnico de Leiria
Luis Paulo Reis	Universidade do Minho
Luis Paulo Santos	Universidade do Minho
Luis Romero	IPVC/ESTG
Manuel Ferreira	ENERMETER, Sistemas de Medição
Manuel J. Fonseca	Faculdade de Ciências da Universidade de Lisboa
Maria Beatriz Carmo	BioISI-Faculdade de Ciências da Universidade de Lisboa
Mário Rui Gomes	IST Universidade de Lisboa
Maximino Bessa	Universidade de Trás-os-Montes e Alto Douro
Miguel Leitão	INESC TEC / Instituto Superior de Engenharia do Porto
Miguel Melo	INESC TEC
Nuno Correia	FCT Universidade Nova de Lisboa
Nuno Rodrigues	ESTG-Instituto Politécnico de Leiria
Óscar Mealha	Universidade de Aveiro
Paulo Dias	IEETA / Universidade de Aveiro
Paulo Menezes	ISR-UC / Universidade de Coimbra
Pedro Faria Lopes	ISCTE-IUL, ISTAR-IUL
Pedro Miguel Moreira	IPVC & LIACC
Pedro Rangel Henriques	Universidade do Minho
Teresa Chambel	Faculdade de Ciências da Universidade de Lisboa
Teresa Romão	FCT Universidade Nova de Lisboa
Vítor J. Sá	Universidade Católica Portuguesa



Organizing Committee / Comissão Organizadora

Chairs / Presidentes:	Paulo Menezes - DEEC/ISR-UC José Carlos Teixeira - DMat-UC/IT
Demonstrators / Demonstradores:	Lícínio Roque - DEI-UC
Secretariat / Secretariado:	Lara Costa - ISR-UC
Publication / Publicações:	João Quintas - IPN/ISR-UC
Internet / Página Internet:	Bruno Patrão ISR-UC
Finances / Finanças:	Marina Dias ISR-UC
Technical Support / Suporte Técnico:	Luís Almeida IPT / ISR-UC Samuel Pedro ISR-UC José Francisco ISR-UC Tiago Catarino ISR-UC



Organization and Sponsors / Organização e Apoios

Organization / Organização

Instituto de Sistemas e Robótica ISR-UC
<http://www.isr.uc.pt>

Grupo Português de Computação Gráfica
<http://www.gpcg.pt>

Sponsoring / Apoios

Departamento de Engenharia Electrotécnica e Computadores DEEC/UC
<http://www.deec.uc.pt>

Universidade de Coimbra UC
<http://www.uc.pt>

Caixa Greal de Depósitos CGD
<http://www.cgd.pt>

Switch Technologies SWTL
<http://www.swtl.pt>



Contents / Índice

Preface	v
Prefácio	vii
Technical Committee / Comissão Científica	ix
Organizing Committee / Comissão Organizadora	xi
Organization and Sponsors / Organização e Apoios	xiii
Keynote Talks / Palestras	1
Da Ciência da Computação para o Mercado Internacional de Jogos <i>Jeferson Valadares</i>	1
Microscopic Crowd Simulation <i>Julien Pettré</i>	1
Interacting Naturally, Reconstructing and Augmenting our World <i>Miguel Sales Dias</i>	1
Interaction Design	3
Escultura Présence: um sistema tridimensional de interação sonoro-corporal <i>Rudolfo Quintas, Adérito Marcos, Mirian Tavares</i>	4
Design de Funções Transferência para Imagens Médicas 3D recorrendo a uma Interface baseada em Esboços <i>Pedro F. Parreira, Ana R. Mendes, Daniel Simões Lopes, Joaquim Jorge</i>	13
Enabling low-complexity devices for interaction with 3D media content via Android API <i>Ricardo Santos, Hugo Costelha, Luis Bento, Pedro Assunção, Márcio Barata</i>	20
Acesso Multimodal em Dispositivos Móveis a Vídeo Georeferenciado através da Forma, Velocidade e Tempo <i>Sérgio Serra, Ana Jorge, Teresa Chambel</i>	24
Visualização tri-dimensional interativa de informação demográfica mundial <i>Eduardo Duarte, Pedro Bordonhos, Paulo Dias, Beatriz Sousa Santos</i>	31
Virtual, Mixed and Augmented Reality	39

How to Deal with Motion Sickness in Virtual Reality <i>Bruno Patrão, Samuel Pedro, Paulo Menezes</i>	40
Incremental Reconstruction Approach for Telepresence or AR Applications <i>Luís Almeida, Paulo Menezes, Jorge Dias</i>	47
Modelo de descrição de experiências multissensoriais multiutilizador <i>Rui Rodrigues, Eduardo Magalhães, Hugo da Silva, Maximino Bessa, António Coelho, Augusto Sousa</i>	55
Interactive configurable virtual environment with Kinect navigation and interaction <i>João Pinto, Paulo Dias, Sérgio Eliseu, Beatriz Sousa Santos</i>	63
Visualização de Objetos Off-Screen em Realidade Aumentada Móvel <i>Gonçalo Silva, Maria Beatriz Carmo, Ana Paula Afonso, Ana Paula Cláudio, António Ferreira</i>	71
Interactive Computer Graphics	79
Visualização de Fluxos Migratórios: Os Resultados das Candidaturas ao Ensino Superior Público <i>Tiago Brito, Joaquim Madeira, Beatriz Sousa Santos</i>	80
Modelação em Superfícies Interactivas para Fabricação 3D <i>Daniel Mendes, Pedro Pascoal, Miguel Nixo, Alfredo Ferreira</i>	88
Patient Specific Implants Design Tool for Surgeons <i>Gustavo Mações, Nuno Sousa, Nelson Alves, Nelson Fânzeres, Rui Coelho</i>	96
Mid-Air Manipulation of 3D Models in (Semi-)Immersive Virtual Environments <i>Vasco Rodrigues, Daniel Mendes, Alfredo Ferreira, Joaquim Jorge</i>	100
Computer Vision & 3D	109
Reconstrução de Superfícies Trianguladas a partir de Nuvens de Pontos sem Restrições Angulares <i>Gonçalo Leitão, Abel Gomes</i>	110
A GPU Approach to Augmented Reality using Non-Central Catadioptric Cameras <i>Tiago Dias, Pedro Miraldo, Nuno Gonçalves</i>	117
Especificação de Interrogações para Recuperação 3D em Ambiente Imersivo <i>Pedro Pascoal, Daniel Mendes, Alfredo Ferreira, Miguel Sales Dias</i>	125
Malhas Segmentadas com Resolução Variável <i>Rui S. V. Rodrigues, José F. M. Morgado, Abel Gomes</i>	133
Uma aplicação móvel para implementar o conceito de caderneta de cromos aos Museus <i>Luís Costa, António Coelho</i>	141
Posters	149
A context-aware immersive interface for teleoperation of mobile robots <i>João Quintas, Luís Almeida, Elísio Sousa, Paulo Menezes</i>	150
3D and the Web - The last twenty years and the future	156
Visualizações Interactivas de Trajectórias em Vídeo no Espaço e no Tempo <i>Ana Jorge, Sérgio Serra, Teresa Chambel</i>	163
Author Index / Índice de Autores	171



Keynote Talks / Palestras

Da Ciência da Computação para o Mercado Internacional de Jogos

Jeferson Valadares

Nesta sessão, Jeferson irá discutir a sua experiência no mercado internacional de Games. Bacharel e Mestre em Ciência da Computação, ambos focando em Games e Inteligência Artificial, Jeferson ingressou na indústria em 2000 através da fundação de sua própria empresa. Apesar de ter trabalhado em diversas plataformas ao longo dos anos, ele tem se focado em Mobile Games desde 2005. Em sua carreira, Jeferson trabalhou em vários títulos que foram sucesso de crítica e de vendas, assim como títulos para marcas como Dragon Age, Simpsons, EA Sports FIFA, Harry Potter, Need for Speed e vários títulos para a Hasbro, como Game of Life e Battleship. Atualmente na BANDAI NAMCO, trabalha com marcas como PAC-MAN, Dark Souls e Dragon Ball.

Short Biography

Jeferson Valadares é VP de Desenvolvimento de Produto na BANDAI NAMCO Entertainment America. Brasileiro, ele é Bacharel em Mestre em Ciência da Computação, e começou a sua carreira na área de games em 2000 através da fundação de sua própria empresa. Antes de ir para a BANDAI ele trabalhou como líder de estúdio na Electronic Arts, BioWare, Flurry, Playfish e Digital Chocolate. Desde 2005 ele deixou o Brasil, e nos últimos dez anos morou em Helsinki, Londres, e agora reside em San Francisco.

Microscopic Crowd Simulation

Julien Pettré

This presentation introduces the main principles of microscopic crowd simulation. Microscopic approaches compute the motion at large scale of a crowd made of dense and numerous entities which results from the combination of the many local interactions between individuals. We explain the recent advances in the field, and more specifically some approaches based on artificial vision which give an attempt to reproduce the perception / action loop that drives human locomotion in dynamic environments. We finally present some applications for crowd simulation to animate large populations in large-scale interactive virtual worlds. We show how to adapt a crowd simulator to allow animating crowds beyond the limitations set by the complexity of crowd simulation algorithms and the available computational resources.

Short Biography

Julien Pettré is research scientist at INRIA since 2006. He obtained a PhD in Robotics in 2003 from the University of Toulouse III in France, prepared under the direction of Jean-Paul Laumond. He then obtained a Marie-Curie IEF grant to spend a 18 months post doc at VRlab, EPFL, Switzerland, headed by Daniel Thalmann. Julien Pettré started his research activities on motion planning for digital humans moving in their virtual environment. He then took interest in virtual crowds. He designed efficient crowd animation techniques to massively populate virtual worlds. He studied real humans moving in crowds in the aim of designing realistic simulation algorithms as well as to provide evaluation methodologies. He finally takes interest in Virtual Reality as an efficient research tool to study human behaviors by setting them in interaction with virtual crowds.

Interacting Naturally, Reconstructing and Augmenting our World

Miguel Sales Dias

In this presentation I'll be addressing the current experiments that myself and my team are carrying in the area of Human-Computer Interaction, Virtual and Augmented Reality, focusing on ways to reconstruct, classify and augment our living and working environment, and on how to naturally communicate with such environments, using natural and multimodal HCI modalities and their fusion, such as speech, gesture (including gesture for sign language recognition), gaze and biometry, this last one used to derive human emotional states. I'll be illustrating the use of such multimodal HCI in augmentative communication, architecture design and ambient assisted living scenarios.

Short Biography

José Miguel Sales Dias holds a bachelor (1985) and a master (1988) in Electrical and Computer Engineering (IST-UTL, Portugal) and a PhD in Computer Graphics and Multimedia (1998) from ISCTE-IUL where he was an Associated Professor until 2005, holding currently an Invited Associated Professor position, teaching and conducting research in Computer Graphics, Virtual and Augmented reality, Ambient Assisted Living and Multimodal Human-Computer Interaction. Since November 2005, he is the Director of the first European R&D Centre in Speech and Natural User Interaction Technologies of Microsoft Corporation in Portugal (Microsoft Language Development Center, MLDC, <http://www.microsoft.com/pt-pt/mldc/>). He is regularly commissioned by the European Commission for R&D project evaluations and reviews. Author of 1 patent, author, co-author or editor of 11 scientific books or journal editions, 12 indexed papers in international journals, 26 chapters in indexed international books, 144 other publications, workshops or keynotes in international conferences. Since 1992 he has participated or participates in 33 International R&D projects (ESPRIT, RACE, ACTS, TELEMATICS, TEN-IBC, EUREKA, INTERREG, FP5 IST-IPS, FP6 IST, ESA, Marie Curie, AAL, ACP), and 15 National (FCT, QREN, NITEC, POSC, POCTI, POSI, ICPME, TIT). He obtained 5 scientific prizes. He is a member of ACM SIGGRAPH, Eurographics, ISCA and IEEE; editorial boards of several journal; several Program Committees of National and International conferences in Computer Graphics, Virtual and Augmented Reality, Speech technologies, Accessibility and Ambient Assisted Living. He was President of ADETTI, an ISCTE-IUL associated R&D research center. He was Vice-president and Secretary of the Portuguese Group of Computer Graphics, Eurographics Portuguese Chapter. Born at December 1st 1961, happy father of three children, two girls and one boy.



Interaction Design

Contents

Escultura Présence: um sistema tridimensional de interação sonoro-corporal <i>Rudolfo Quintas, Adérito Marcos, Mirian Tavares</i>	4
Design de Funções Transferência para Imagens Médicas 3D recorrendo a uma Interface baseada em Esboços <i>Pedro F. Parreira, Ana R. Mendes, Daniel Simões Lopes, Joaquim Jorge</i>	13
Enabling low-complexity devices for interaction with 3D media content via Android API <i>Ricardo Santos, Hugo Costelha, Luis Bento, Pedro Assunção, Márcio Barata</i>	20
Acesso Multimodal em Dispositivos Móveis a Vídeo Georeferenciado através da Forma, Velocidade e Tempo <i>Sérgio Serra, Ana Jorge, Teresa Chambel</i>	24
Visualização tri-dimensional interativa de informação demográfica mundial <i>Eduardo Duarte, Pedro Bordonhos, Paulo Dias, Beatriz Sousa Santos</i>	31

Escultura Présence: um sistema tridimensional de interação sonoro-corporal

Rudolfo Quintas±, Adérito Marcos±, Mirian Tavares*

± Centro de Investigação em Artes e Comunicação, Universidade Aberta, Lisboa, Portugal *Universidade do Algarve, Faro, Portugal

(rudolfo.quintas@uab.pt, aderito.marcos@uab.pt, mtavares@ualg.pt)

Resumo

Este artigo descreve o sistema que suporta a escultura Présence, trata-se de um sistema composto por uma aplicação de captura de movimento tridimensional (Localização 3D) e uma aplicação de som que possibilita uma experiência sonora interativa e imersiva através da expressão corporal. O sistema foi desenvolvido para ser utilizado por um público geral, heterogéneo, não especializado quer em tecnologia ou composição sonora. Os utilizadores colocam uns auscultadores sem fios e quando entram numa determinada área sensível iniciam a audição de uma composição sonora criada pelo movimento do seu próprio corpo. Esta composição permanece em constante modelação e organização de acordo com o movimento do utilizador no espaço. Este funcionamento ocorre através da leitura de um conjunto de posições tridimensionais do corpo do utilizador mapeados como parâmetros de controlo para um conjunto de geradores de som. O sistema foi testado em sete exposições públicas no qual participaram e interagiram mais de 500 pessoas comprovando a flexibilidade e expressividade sonora do sistema.

Palavras-chave

Interfaces Naturais, Captura de Movimento 3D, KINECT, Composição Sonora Interativa, Arte Sonora, Performance, Expressão Corporal.

1. INTRODUÇÃO

Podemos definir ‘*Sound Art*’ ou Arte Sonora como uma prática artística que materializa um pensamento artístico através do som. Como termo e prática artística, a Arte Sonora consolidou-se e cresceu exponencialmente desde o final da década de 90, manifestando-se através de várias exposições em galerias e museus, no entanto, foi na década de 60 que o termo surgiu e se afirmou (Cox 2009). Brandon Labelle descreve uma nova tendência na Arte Sonora relacionada com as *networks* e os *digital media* (Labelle 2006), que na sua perspectiva, formalizam e expandem o conceito de Estética Relacional (Bourriaud 2002) no domínio do som. Os meios digitais, não só se refletiram no dia a dia do ser humano em geral como também na criação artística, uma vez que o computador se tornou o estúdio e a ferramenta para um grande número de novos artistas, e o próprio contexto digital, local ou em rede, social ou laboratorial, uma fonte de inspiração, reflexão ou subversão. Esta tendência repercutiu-se também na Arte Sonora como é o caso das instalações *Global Strings* (1998-) de Atau Tanaka e Kasper Toepfritz, *Sound Grid* (2002) de Achim Wollscheid ou a instalação sonora interativa *Very Nervous System* (1986-1990) de David Rokeby, pioneira na relação de criação de música através do movimento do corpo (Cooper

1995). Por sua vez, o computador como principal ferramenta de criação impulsionou a geração “*Lap-top*” que explorou a capacidade do computador como ferramenta de composição e performance, artistas e músicos como Scanner (1964), Ryoji Ikeda (1966), Carsten Nicolai (1965), @c + Lia, Richard Chartier (1971) ou Ryoichi Kurokawa (1978, JP), partilham em comum o mundo do minimalismo estético e do “*Pós-Digital*” (Kascone 2000), e o facto de quase todos transitarem entre as salas de concertos e as galerias, demonstrando assim que as relações entre o mundo da música experimental e da arte continuam presentes nas relações da Arte Sonora no séc. XXI. Como meio de expressão plástica a Arte Sonora está atualmente integrada no circuito de arte contemporânea. São exemplos do passado recente o *Turner Prize Award 2010*¹ entregue a Suzan Philips (1965) pela instalação sonora *Lowlands* (2010) ou alguns dos recentes projetos do artista francês Célest Boursier-Mougenot como o “*Here to Ear*” no museu PEM².

¹ Prémio estabelecido em Inglaterra, em 1984, para artistas plásticos britânicos com menos de 50 anos, com o objetivo de celebrar novos desenvolvimentos em arte contemporânea.

² https://www.youtube.com/watch?v=Yn93J2axD_k

Neste artigo apresentamos o *Présence*, uma escultura mediada por computador resultante de um projeto artístico que explora a criação e representação de retratos sonoros e corporais dos utilizadores - visitantes e participantes numa exposição ou galeria de arte. Os retratos são coreografias espontâneas e composições sonoras electrónicas geradas e registadas em tempo real através da interação afectada à experiência de cada participante (Quintas, Marcos e Mirian 2014), ver Figura 1.



Figura 1. Imagem de captura de ecrã de um vídeo que documenta a interação de alguns visitantes³ com o sistema *Présence*.

A escultura integra um sistema de composição sonoro interativo que visa proporcionar experiências sensoriais e participativas aos utilizadores ao nível da audição e do movimento do corpo. Essa experiência é sonora, no domínio da composição de música electrónica, e corporal, através do movimento e gestos dos utilizadores (tais como posição espacial do corpo, levantar braços, baixar o corpo) expandindo uma dimensão performativa e coreográfica que gera uma auto representação (auto retrato sonoro). Do ponto de vista artístico é um trabalho que se insere na linguagem das artes plásticas, no domínio da média-arte digital, operando nas fronteiras das disciplinas da Arte Sonora e Performance.

O artigo está organizado da seguinte forma: na secção 2 descrevemos investigação relacionada e o sistema de interação sonoro que suporta a escultura *Présence*; na secção 3 descrevemos as experiências com utilizadores obtidas ao longo de vários testes com o sistema resultantes de sete exposições públicas onde a escultura foi apresentada; Concluimos na secção 4 sobre as intenções iniciais e o trabalho desenvolvido apontando ideias futuras..

2. O SISTEMA DE INTERAÇÃO SONORO DO PRÉSENCE

Projetos artísticos no campo das instalações sonoras que mapeiam gestos e movimento para som como o trabalho pioneiro de David Rockbey "*Very Nervous System*" (1986-1990) (Cooper 1995), a instalação sonora "*Anything Can Break*"⁴ (2012) de Pinaree Sanpitak criada

para a Bienal de Sidney de 2012, ou "*KlangDerWisch: gesture-driven sound installation*"⁵ (2015) de Karlheinz Essl, e a investigação no campo da interação sonoro-musical, têm demonstrado como traduzir movimentos abstratos e informação gestual em regras mais universais sobre as emoções e afectos como é o caso da pesquisa pioneira de Camurri com "*Eyes Web*" (Camurri et al. 2000) no contexto do Kansei (Camurri et al. 2002) ou a pesquisa de Arfib et al. (2002) que propões modelos de abstração entre gestos e algoritmos de síntese sonora.

Estes exemplos são referências importantes para a pesquisa e criação comunicada neste artigo porque demonstram como os utilizadores interagem em instalações públicas ou desenvolvem modelos de leitura de emoções através de movimentos. O trabalho comunicado neste artigo contribui para a área do mapeamento de gestos e movimentos para som em instalações de arte por ser:

- Um sistema de interação sonora-musical tridimensional orientado para o visitante de uma galeria ou museu;
- A composição sonora e mapeamentos orientados ao design da percepção do público em geral não especialista em criação sonoro-música e corporal (dança);
- Explorar a ideia de um auto-retrato pela autorrepresentação ao vivo que gera uma criação sonora que é registada num ficheiro áudio.

Assim, a investigação e desenvolvimento deste sistema sonoro interativo foi orientado para a criação de uma interface intuitiva e responsiva vocacionada para o público em geral. O objectivo é que as pessoas possam viver uma experiência sonora original, expressiva, mediada por computador, sem a necessidade de possuírem um conhecimento prévio em composição musical ou aprendizagem do interface. Esta orientação coloca um conjunto de características e desafios distintos do desenvolvimento de um instrumento musical ou da criação de uma nova interface de expressão musical (NIME) como por exemplo o protótipo WAVE (Valbom e Marcos 2007) ou o Glitch Delighter (Quintas 2010).

2.1 Requisitos de utilização

O desenvolvimento de um instrumento musical é tipicamente orientado para ser testado por um utilizador com conhecimentos musicais ('utilizador-músico'). Por norma a aprendizagem de um novo instrumento acarreta um tempo de exploração com a duração mínima de algumas semanas até que o corpo tenha memorizado os novos gestos que permitam manipular o instrumento com fluidez para que posteriormente a atenção seja concentrada na música, ou seja no escutar e no sentir. Não obstante o sistema aqui desenvolvido poder ser tocado por um músico, a metodologia de desenvolvimento foi orientado por

³ Disponível para visualização em: <https://www.youtube.com/watch?v=PMaUHHCq66U#t=42>

⁴ Disponível para visualização em: <https://vimeo.com/47503996>

⁵ Disponível para visualização em: <https://www.youtube.com/watch?v=39zlwZ837Gc>

um conjunto de princípios distintos porque a aprendizagem teria que ser a) imediata, b) não esbater em dificuldades técnicas e 3) transmitir um crescendo na experiência estética (evolução cognitiva, emocional e sensorial). Ou seja, pretende-se que a aprendizagem do ‘utilizador-público geral’ seja realizada de forma natural levado o utilizador a sentir envolvimento na experiência relativamente ao que está a ouvir sem necessitar de ser forçado a passar por um processo de aprendizagem prévia do instrumento/sistema. Para que o prazer sensorial e envolvimento na experiência fosse alcançado de forma imediata, esta teria que ser intuitiva, tendo em conta que

(1) todo o processo de composição sonora teria de ser estruturado com base na dedução de determinadas ações, gestos e medidas antropométricas dos utilizadores consideradas previsíveis, e

(2) a interface deveria permanecer invisível enquanto o utilizador fosse aprendendo interações complexas de forma intuitiva de forma não necessariamente consciente delas, características tipo do desenvolvimento interfaces naturais ou NUI (*Natural User Interfaces*).

Assim, o desenvolvimento e testes de utilizador foram realizados com base no convite a um público não especializado, portanto não dirigido a músicos, que experimentaram o sistema, o que permitiu, através da observação dos seus comportamentos, que o sistema fosse gradualmente afinado. A Tabela 1 resume um conjunto de características comparativas entre o instrumento musical propriamente dito e o artefacto sonoro-musical interativo com um propósito artístico experimental e aberto:

	Instrumento	Artefacto Interativo
Conceito do sistema	Baseado em famílias de instrumentos e no que acrescenta.	Serve uma intenção artística (ler a intenção artística na introdução).
Originalidade	Baseado em características distintivas relativamente a outros instrumentos da mesma categoria.	Na forma de realizar a intenção artística (ler a intenção artística na introdução).
Utilizador	Músico	Público Geral
Curva de Aprendizagem	Lenta: na ordem dos dias, semanas, meses...	Rápida: Imediato, intuitivo.
Objectivo	Tocar uma pauta, improvisar, instrumento de composição, composição.	Realizar uma experiência.
Interface	Tendencialmente associado a gestos de famílias de instrumentos, como precursão, cordas, sopro, etc.	Procura de NUI – <i>Natural User Interfaces</i> .
Gestos	Gestos de grande complexidade e sequência. Micro-gestos: dedos, movimentos precisos e detalhados.	Gestos simples. Macro-gestos: movimento do corpo, movimentos imprecisos como levantar braços, baixar o corpo. Ações e gestos Previsíveis baseados no desenho da experiência.

Composição	Em função do compositor.	Em função do utilizador: esta é uma das características mais distintivas pois um artefacto expositivo já contém uma composição pré-definida e estruturada e a qualidade do mesmo depende também da composição que lhe é associada.
Metodologia, testes de desenvolvimento	Testes baseados na observação e inquérito a músicos a tocar o instrumento.	Testes baseado na observação do público em geral a experimentar a instalação.
Expressividade	Depende da qualidade dos geradores sonoros e dos mapeamentos aplicados.	Depende da qualidade dos geradores sonoros e dos mapeamentos aplicados e do desenho da experiência.
Principal factor qualitativo da experiência	A qualidade interface e dos sons gerados.	Se comunica a intenção artística com clareza.

Tabela 1: Características comparativas na abordagem à criação de um instrumento e um artefacto interativo sonoro-musical.

2.2 Visão Geral do Sistema

O sistema incorpora uma câmara que reconhece o corpo dos participantes gerando um modelo simplificado do esqueleto do corpo. Algumas das coordenadas tridimensionais (3D) do esqueleto tais como a posição das mãos ou localização do tronco no espaço são processadas e mapeadas para um gerador de som que envia os sons produzidos para os auscultadores que o participante está a utilizar. Ver figura 2 e 3. Na figura 2 podemos observar um aspeto de um utilizador a interagir com o sistema. Este encontra-se dentro da área de ativação, com os auscultadores colocados, levantando o braço direito, a ouvir o som gerado por esse movimento. À sua frente encontra-se o objecto escultórico preto com a volumetria de um trapézio que se rebate no chão através da linhas de contorno pretas. Na figura 3 é apresentado um esquema sucinto do fluxo de informação do sistema. No interior do objecto escultórico preto encontra-se todo o hardware/software responsável pela captura de movimento e geração sonora que é enviada sem fios para os auscultadores do utilizador.

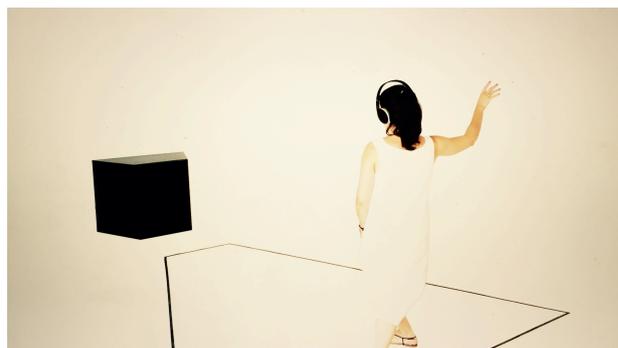


Figura 2. Momento de um utilizador a interagir com a escultura.

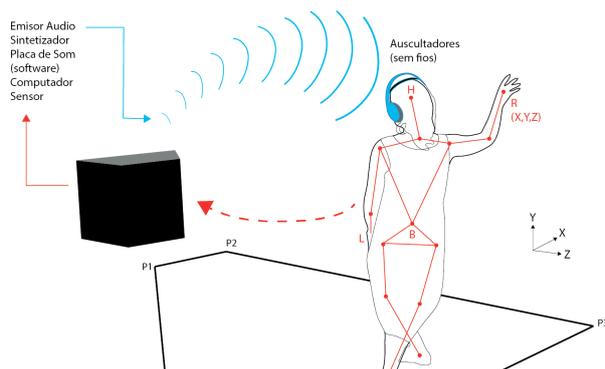


Figura 3. Esquema geral do fluxo de informação do sistema.

Este sistema sonoro possibilita uma experiência imersiva e espacial, fornecendo aos utilizadores um controlo intuitivo e gestual sobre o som que produzem. Através da captura de movimento do utilizador são calculadas posições das principais articulações do corpo, permitindo uma calibração simples da área de interação assim como do mapeamento das articulações do esqueleto do corpo desejáveis a serem mapeadas para o sistema sonoro. A composição sonora é gerada pelo movimento do corpo através da manipulação de sons previamente gravados (*sound samples*) em combinação com a geração de sons puramente sintetizados, permitindo a criação de uma composição infinitamente variável transmitindo a sensação que o som “se esculpe” através do movimento do corpo no espaço. Por último, a composição sonoro-musical gerada durante a utilização do sistema pode ser gravada em ficheiro áudio. Esta representa um registo sonoro único do utilizador em causa – o seu auto-retrato sonoro.

2.3 Hardware e estruturas físicas do sistema

O sistema é baseado numa arquitetura que integra um computador *Mac* da *Apple* com requisitos mínimos de processador 2.6GHz *Dual-Core Intel i5*, um sensor *KINECT VI*, uma placa de som externa com interface *MIDI* com duas entradas e duas saídas áudio, um processador de efeitos/sintetizador *KORG Kaos PAD 3*⁶, um conjunto de auscultadores sem fios e o respectivo emissor áudio. Finalmente todo o hardware é acondicionado dentro do objecto escultórico preto cujas estruturas foram construídas em madeira e o acabamento em acrílico preto. Este objecto escultórico desdobra-se no chão através da expansão bidimensional do desenho da sua base inscrito no chão, compondo um espaço plástico delimitado ao mesmo tempo que aberto, concreto e imaterial, que se complementa e finaliza com a entrada do corpo, ver Figuras 4 e 5.



Figura 4. Vista interior do objecto escultórico que acondiciona todo o hardware/software do sistema interativo.



Figura 5. Três diferentes vistas do objecto escultórico fechado após acondicionamento do hardware.

2.4 Arquitetura do Sistema

Do ponto de vista da experiência do participante o sistema transforma gestos e movimentos em som. Mas do ponto de vista informático o sistema tem como objetivo a conversão de uma imagem de profundidade em som. Essa imagem em relevo é gerada por um sensor de profundidade *KINECT* e processada pelo software de captura de movimento *NI MATE*⁷. Este software faz a detecção do utilizador na imagem e calcula as pseudoposições tridimensionais do esqueleto do corpo do utilizador. O resultado deste processo é descrito na subsecção 2.5.1 como “Sistema de Localização” cujo *output* é o envio de uma lista de coordenadas 3D expressas em milímetros no espaço através do protocolo *Open Sound Control (OSC)*. O sistema de áudio descrito na subsecção 2.5.2 recebe as coordenadas tridimensionais e tem como objetivo o mapeamento e conversão destas em escalas que interpretadas por diferentes módulos como estados ou parâmetros de geradores de som.

⁷ <http://www.ni-mate.com>

⁶ http://www.korg.com/us/products/dj/kaoss_pad_kp3_plus/

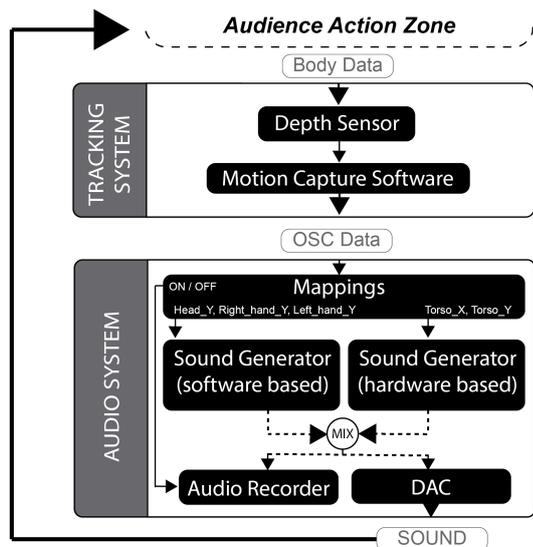


Figura 6: Arquitetura do sistema Présence

A Figura 6 ilustra a arquitetura do sistema Présence onde podemos observar os dois módulos nucleares principais, sistema de áudio e sistema de localização ou *tracking* e o fluxo informativo entre estes e o exterior, cujo processamento passa por três estados distintos: no primeiro o utilizador encontra-se numa área de ação onde é recolhida a imagem de profundidade <Body Data> que, no estado seguinte, é convertida em posições tridimensionais em milímetros no espaço <OSC Data> e posteriormente mapeadas para gerarem o <som>.

2.4.1 Sistema de Localização

NI-MATE é o software responsável pela aquisição da imagem do sensor de profundidade (*KINECT*) e pelo cálculo e geração do pseudo-esqueleto tridimensional do corpo em tempo-real, ver figura 7.

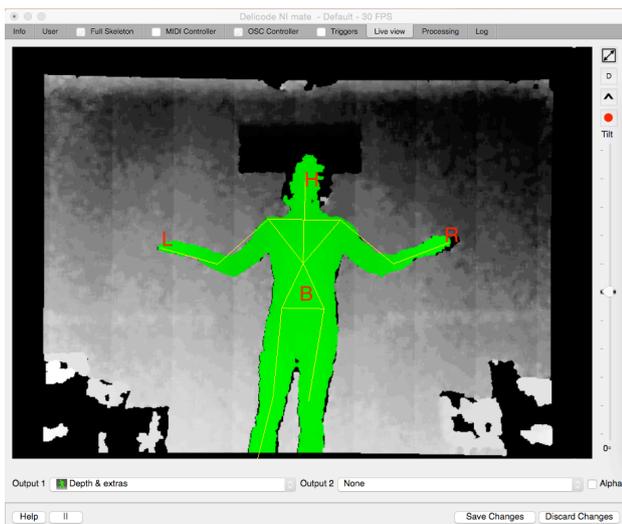


Figura 7: Captura do ecrã da aplicação NI-MATE onde se observa a verde o utilizador ativo e o desenho do esqueleto do corpo do utilizador.

Modo de Ativação: O software foi configurado e calibrado para que o utilizador fique ativo, isto é, para que sejam calculadas as coordenadas tridimensionais do esqueleto do corpo para o utilizador que estiver mais próximo do sensor. Isto facilita a experiência porque não é necessário dar uma instrução previa ao utilizador como por exemplo fazer uma determinada pose, levantar um braço ou ficar parado num determinado lugar para se tornar um utilizador ativo. Este método introduz um ligeiro ruído no sistema, uma vez que se outro utilizador se mover mais próximo do sensor este passa a ser o utilizador ativo. No entanto, como a experiência é individual e realizada em conjunto com os auscultadores este problema não se coloca.

Área de Ativação: O processamento e geração do esqueleto tridimensional do utilizador foi configurado para ser calculado apenas quando o utilizador se encontra dentro de uma determinada área. Neste caso corresponde à área de um trapézio demarcada no plano do chão e visível para o utilizador pela demarcação do contorno, ver Figuras 2 e 3. Esta área é geometricamente descrita pelos seguintes pontos da Tabela 2:

	x	z
P1	-300	800
P2	300	800
P3	1400	3500
P4	1400	3500

Tabela 2: medidas em milímetros, onde o ponto (0,0,0) é o sensor.

2.4.2 Sistema de Áudio

O sistema áudio foi desenvolvido em Max⁸ que é uma linguagem de programação gráfica multimédia orientada por objetos, iniciada nos anos 80 para controlo *MIDI* e processamento sinal de áudio atualmente expandida para imagem e 3D. Atualmente é das linguagens multimédia mais utilizadas por criativos em todo o mundo que trabalham a interatividade e sistemas multimédia em tempo-real. No Max 7 foi programada uma aplicação que implementa 4 módulos:

- M1) mapeamentos: sistema recepção e processamento dos valores OSC relativos às coordenadas do esqueleto do corpo enviados pelo software NI-MATE que depois de mapeados servem como controladores para os geradores de som;
- M2) um gerador de som, responsável pelo playback e processamento de efeitos de ficheiros áudio;
- M3) um interface de comunicação MIDI com o processador de efeitos/sintetizador exterior (o KORG KAOS PAD 3);
- M4) um gravador de áudio;

⁸ <https://cycling74.com>

M1 Mapeamentos:

“The mapping “design” [...] Choosing how <one> controller will be mapped to <one> synthesis parameter (sound generator) is a matter of sensibility. No math or general technique is involved, no right or wrong, just more or less expressive it can result.” (Quintas 2010)

O mapeamento é uma espécie de ponte que une duas realidades distintas que pretendem comunicar entre si, por norma de forma unidirecional. É um assunto central em todas as formas de interação. Faz a ponte entre o utilizador e os materiais onde este atua. Num sistema de sonoro-musical interativo estabelece a ligação essencial que faz expressar as ideias sobre a forma e a estrutura da composição (Schacher 2010).

O mapeamento divide-se em duas tarefas: uma técnica e outra artística. Neste sistema a técnica corresponde ao conhecimento de conversão de valores da escala do espaço-tempo-corpo (milímetros, posições, velocidades) em escalas que façam sentido como parâmetros para os geradores e controladores de som (volumes, ataques, escalas, eventos, entre outros). Por exemplo, as variáveis do esqueleto correspondem a coordenadas tridimensionais no espaço e antes de serem utilizadas como valores nos controladores dos geradores de som são mapeadas para valores que fazem sentido como parâmetros dos respectivos controladores: assim o mapeamento da profundidade do corpo no espaço que dentro da área de ativação tem como valor mínimo 800 milímetros e valor máximo 3500 milímetros é mapeado para corresponder a valores entre 0 e 127 - valores da escala MIDI. A tarefa artística respeita a realização da composição idealizada pelo compositor e o design da percepção: a concepção de como um determinado som é percebido pelo movimento de um gesto.

O mapeamento em sistemas interativos de áudio em tempo-real encontra o seu maior desafio nas instalações de arte sonoras: um visitante típico numa instalação interativa tem menos consciência e sensibilidade para os aspectos de movimento espaciais e físicas dos elementos do corpo, movimento, espaço e esforço, do que um músico ou dançarino. Isto significa que toda a estrutura musical tem de ser composta para evoluir numa experiência perceptual de adaptação crescente - a estratégia artística foi a de depurar as possibilidades de interação simplificando-as ao limite, depurando-as, mantendo uma noção de corpo-espaço e corpo-gestos. Isto também significa que os mapeamentos tem que ser concebidos de modo a expandirem e aumentarem a expressão do corpo do visitante, ver figuras 10,11, e 12 a título de exemplo. A composição tem de ser estruturada numa sequência onde o participante intuitivamente entende que há muitas camadas para explorar. Os princípios que regeram estas ideias encontram-se nos em “2.1 requisitos do sistema”: entre outros, para ser utilizado por um público geral, heterogéneo, não especializado quer em tecnologia ou composição sonora.

Assim o mapeamento entre gestos/movimentos de partes do corpo e os efeitos sonoros foram desenhados segundo

o princípio de “*less is more*” no sentido de depurar quais as partes do corpo consideradas suficientes para possibilitar uma experiência corpo-espaço-som e corpo-gestos-som. O objetivo foi o de simplificar as relações de mapeamento para encurtar o tempo de aprendizagem e permitir maior fluxo na experiência.

A inspiração para a simplificação destas relações encontra-se na teoria de Laban sobre a distinção entre a análise do movimento no “*Personal Space*” conhecido com *Kinesphere* e a análise do movimento no “*General Space*” (Laban 1963). Em “*Model Educational Dance*” (Laban 1963, p.85) Laban escreve: “*Whenever the body moves or stands, it is surrounded by space. Around the body is the sphere of movement, or Kinesphere, the circumference of which can be reached by normally extended limbs without changing one’s stance, that is, the place of support. The imaginary inner edge of this sphere can be touched by hands and feet, and all points of it can be reached. Outside this immediate sphere lies the “general” space, in which the human can enter only by moving away from his/her original stance. He/she has to step outside the borders of his immediate sphere and create a new one from the new stance, or, in other words, he transfers what might be called his “personal” sphere to another place in the general space. Thus, in actual fact, he never goes outside his personal sphere of movement, but carries it around with him like a shell.*”

Em *Présence* a ideia para o mapeamento do corpo no espaço (“*General Space*”) foi construído com a localização e seguimento da posição do centro do corpo “*Torso_X*” e “*Torso_Y*” sendo mapeado para a modelação de síntese sonora em tempo real de um conjunto de frequências e efeitos sonoros abstratos que acompanham o utilizador no espaço com um estilo sonoro “*Drone*”⁹. Esta abstração sonora em *Drone* é uma ideia de composição para transmitir a sensação que ao deslocar-se no espaço o participante está a mover o som. Esta foi a primeira estrutura da composição a ser desenvolvida. Através da experiência de observação de vários utilizadores, a primeira ação que fazem é movimentarem-se no espaço. Só depois é que procuram outras situações como mover partes do corpo - o “*Personal Space*”.

O mapeamento do “*Personal Space*” foi simplificado para três ações: movimentar os braços na vertical e baixar o centro de gravidade do corpo. Os braços foram escolhidos porque são os membros que permitem maior controle sobre expressividade e velocidade uma vez que nem todas as pessoas têm equilíbrio e destreza física para levantar ou baixar as pernas com velocidade.

Era importante manter uma relação de exploração de gestos confortáveis para minimizar eventuais bloqueios com limitações físicas. Assim, foi utilizado o mapeamento da mão esquerda “*Left_hand_Y*” e direita “*Right_hand_Y*”

⁹ Um estilo musical minimalista que realça o uso de sons sustentados ou repetidos, notas ou clusters de tons.

“a sensação de liberdade como se estivessem num espaço de som sem gravidade”, outros pela forma como “o som se tornou um indutor do espaço e do movimento”, outros pela forma como “a abstração do som suscita a criação de imagens mentais” e “pelo o sentimento de que a experiência se torna muito pessoal”, outros ainda que ao experimentarem de olhos fechados “sentiram como se estivessem num sonho”, ver Figura 9.



Figura 9: Captura do ecrã de vídeo que documenta a experiência de um conjunto de utilizadores. Disponível em: <https://www.youtube.com/watch?v=L-NHtyQYx0I>

A diversidade de pontos de vista sobre a experiência afecta a cada pessoa é uma resposta enriquecedora e qualitativa relativamente à experiência que a escultura acrescenta em cada indivíduo. Não existem duas frases iguais sobre a forma como cada pessoa sentiu a experiência, porque são elas próprias a gerar o sentido último da sua vivência – quanto mais dão de si mais recebem. A experiência é possibilitada por um sistema tecnicamente sofisticado no entanto é da ordem da poética.

Para além do gosto, se gostamos ou não de uma determinada obra/experiência artística, o resultado de tal encontro, sendo o mais importante, está no que ela nos acrescenta, como nos transforma, e o que nos devolve na forma de ver ou experimentar a realidade, o que nos faz descobrir. A experiência artística é assim um canal de abertura para o mundo que se experimenta no exterior mas que se constrói a partir do nosso interior. As opiniões, pensamentos e relatos orais dos participantes revelam tais características por serem singulares a cada indivíduo – como que se a escultura tivesse sido criada a pensar em cada um deles. Essa é a sensação e opinião que transparece quer nas coreografias espontâneas de cada participante, quer nas opiniões gentilmente cedidas e registadas em vídeo que constataam numa outra linguagem, a oral, aquilo que conseguimos observar na linguagem corporal de cada um dos participantes. A ideia de retrato é assim conseguida porque cada coreografia espontânea e composição sonora correspondente representa uma transposição plástica e poética de cada indivíduo. As seguintes imagens representam três momentos distintos da experiência de três participantes. São imagens que exemplificam a imersão que cada um sente na sua experiência, figuras 10, 11 e 12.



Figuras 10,11, e 12: fotografias de três participantes a vivenciar a escultura sonora na galeria Adamastor Studios.

4. CONCLUSÃO

O artigo descreveu técnica e conceptualmente a escultura interativa PRÉSENCE que foi desenvolvida para ser uti-

lizada por um público geral, heterogéneo, não especializado quer em tecnologia ou composição sonora. Este artigo de cariz técnico centrou-se essencialmente da comunicação do sistema desenvolvido e da experiência proporcionada aos utilizadores.

Em termos gerais a após a observação empírica de mais de 500 utilizadores e dos vários depoimentos recolhidos em vídeo, parece claro que o sistema é intuitivo e expressivo, permitindo a fácil utilização e uma experiência sonora imersiva por um público não especializado independentemente da idade ou conhecimentos prévios em composição musical e tecnologias. Concluímos que os bons resultados alcançados com a experiência dos utilizadores (*user experience*) teve como factor principal uma noção clara dos requisitos de utilização como descritos na secção 2.1 e um desenvolvimento baseado em constantes testes com utilizadores comuns. Baseados nos nossos estudos podemos concluir que o sistema traduz a ideia artística do “retrato sonoro”, induzindo a expressão da singularidade, implicando a possibilidade de explorar a modelação ou construção de um subjetividade por via de uma performance de cada participante.

5. IDEIAS FUTURAS

A escultura foi apresentada em sete exposições em espaços interiores. Uma ideia futura é a criação de um conjunto de outras esculturas com outras formas e cores, cada uma com a sua composição sonora para serem montadas em conjunto numa praça ao ar-livre. Aqui colocam-se novos desafios técnicos e de natureza de estratégias artísticas com passagem do interior e de locais onde o público está predisposto à experiência artística para locais públicos onde é a arte que vai ao encontro do cidadão. Para além de considerarmos o interesse plástico da experiência corporal do cidadão comum, a experiência de tornar este projeto um objecto de arte pública é bastante motivador.

A segunda ideia respeita a utilização do sistema em ambiente de performance num espetáculo realizado por bailarinos-performers invisuais onde a sala de espetáculo a certo momento ficará às escuras, e o público, deixando de ver o corpo dos bailarinos é transportado para uma dimensão paralela à dos invisuais.

6. REFERÊNCIAS

- [Cox 2009] Sound Art and The Sonic Unconscious . Organized Sound 14(1): 19-26 @ 2009 Cambridge University Press. Printed in the UK.
- [LaBelle 2006] Background noise: perspectives on sound art. Continuum International Publishing Group. 316 pp.
- [Bourriaud 2002] Relational Aesthetics. Tradução Simon Pleasance e Fronza Woods (Dijon: les presses du reel, 2002).
- [Cooper 1995] " Very Nervous System : Artist David Rokeby adds new meaning to the term interactive", Wired Issue 3.03 (Mar 1995)
- [Cascone 2000] " THE AESTHETICS OF FAILURE : 'Post-Digital' Tendencies in Contemporary Computer Music, Computer Music Journal 24:4 Winter 2000 (MIT Press).
- [Quintas, R, Marcos, A e Tavares 2015] PRÉSENCE n.1: Escultura Sonora Interativa, In PROCEEDINGS of 7th International Conference on Digital Arts, Artech 2015.
- [Valbom & Marcos 2007] An Immersive Musical Instrument Prototype. IEEE Computer Graphics and Applications. PP 14-19, Published by the IEEE Computer Society 2007.
- [Quintas 2010] Glitch Delighter: Lighter's Flame Base Hyper-Instrument for Glitch Music in Burning the Sound Performance, in Conference Proceedings of NIME – New Interfaces For Musical Expression. Sydney.
- [Schacher, Jan. 2010], “Motion To Gesture To Sound: Mapping For Interactive Dance” in: Proceedings of the 2010 Conference on New Interfaces for Musical Expression (NIME10), Sydney, Australia.
- [Laban R. 1963], Modern Educational Dance, Macdonald & Evans Ltd. London, 1963.
- [Jordà, Sergi, 2001]“New Musical Interfaces and New Music-making Paradigms” in: Proceedings of the 2001 New Instruments for Musical Expression Workshop. Seattle: CHI 2001
- [Camurri, A., Hashimoto, S., Ricchetti, M., Ricci, A., Suzuki, K., Trocca, R., Volpe, G.,] “EyesWeb: Toward Gesture and Affect Recognition in Interactive Dance and Music Systems”, Computer Music Journal, v.24 n.1, p.57- 69, April 2000
- [Camurri, A., Trocca, R., Volpe, G.,] "Interactive Systems Design: A KANSEI-based Approach" in: Proceedings of the 2002 Conference on New Interfaces for Musical Expression (NIME02), Dublin, Eire
- [D. Arfib , J. M. Couturier , L. Kessous and V. Verfaillie,] "Strategies of Mapping between Gesture Data and Synthesis Model Parameters Using Perceptual Spaces", Organised Sound 7(2): 127–144, 2002 Cambridge University Press. UK

Design de Funções Transferência para Imagens Médicas 3D recorrendo a uma Interface baseada em Esboços

Pedro F. Parreira Ana R. Mendes
Dep. Bioengenharia, IST
Av. Rovisco Pais, 1000 Lisboa
{pedro.f.parreira,
ana.rita.monteiro.mendes}
@tecnico.ulisboa.pt

Daniel Simões Lopes
INESC-ID Lisboa
Rua Alves Redol, 9
1000-029 Lisboa
daniel.lopes@inesc-id.pt

Joaquim A. Jorge
INESC-ID Lisboa
Dep. Eng^a. Informática, IST
Av. Rovisco Pais, 1000 Lisboa
jorgej@tecnico.ulisboa.pt

Abstract

The visual quality of volume rendering images and the contrast between different anatomical structures strongly depends on the type of transfer function that is applied by the volume renderer. While conventional window-icon-menu-pointer approaches refer the user to apply pre-defined transfer functions for isolating specific tissues, these are not always desirable for many clinical and surgical scenarios. This leaves the user with the tedious and time consuming task of manual transfer function design. In order to make this task more natural, we propose a sketch-based interface where touch gestures are used to directly control voxel opacity and colour. We also present new transfer function design approaches that aim to reduce the number of trial and error iterations, which usually are necessary for finding appropriate transfer functions. By following our approach, users were able to more rapidly explore and understand volume data, as the number iterations necessary to obtain a desirable transfer function where reduced. For both transfer function design and volume manipulation tasks users responded more favourably to our sketch-based interface when compared to standard mouse and keyboard controls, as the former approach promotes reduced design iterations and improve the user's overall understanding of the data.

Keywords

Transfer Function, Volume Rendering, 3D Medical Images, Sketch-based Interface

1. INTRODUÇÃO

A análise de imagens médicas tridimensionais (3D) assenta muito na utilização de ferramentas que evidenciam estruturas anatómicas de interesse contidos no interior de um volume de imagens. Nomeadamente, ferramentas que promovem a manipulação das funções de transferência.

As funções de transferência definem a opacidade e a cor dos voxels, convertendo informação escalar de intensidade, em propriedades ópticas bem definidas como a cor. Por outras palavras, a manipulação de funções de transferência consiste num processo análogo à pseudo-coloração de fotografias antigas a preto e branco. A exploração do espaço de design definido pelas funções de transferência é relevante em vários contextos, desde o auxílio ao diagnóstico, passando pelo planeamento cirúrgico até ao ensino da anatomia [McGhee10; Goodman09; Botha12].

Um factor chave para a correcta manipulação destas funções consiste em disponibilizar ao utilizador informação sobre a distribuição das intensidades dos voxels, isto é, do histograma do volume de dados. A

ausência deste tipo de informação visual dificulta em muito a obtenção do resultado desejado [Lundström06].

O facto das funções de transferência serem inerentemente não espaciais (i.e., nem o domínio nem o contradomínio da função são grandezas geométricas) dificulta o processo de evidenciar estruturas anatómicas que, apesar de serem caracterizadas pelo seu lugar geométrico, não são facilmente distinguíveis em termos das intensidades dos voxels [Kindlmann02]. Por outras palavras, cabe ao utilizador a difícil tarefa de associar o domínio da função de transferência (i.e., intensidade de voxel) à estrutura anatómica de interesse. Quando se lida com imagens obtidas por Tomografia Computorizada (CT), esta dificuldade é atenuada devido ao facto das intensidades de imagens CT estarem escaladas (escala de Hounsfield) e poderem ser associadas a tecidos específicos, apresentado um claro contraste entre tecidos duros e moles. Todavia, caso a estrutura anatómica de interesse não se encontre tabelada em funções de transferência pré-definidas, ou se as imagens a serem utilizadas tenham sido adquiridas por Ressonância Magnética (MRI), obriga assim a uma interacção fortemente baseada em tentativa-erro por parte do utilizador. [Bergeest08].

Acresce-se ainda a limitação de que uma pequena alteração na função de transferência culmina, muitas vezes, numa alteração indesejada de opacidade. A rápida superação de tais limitações passa em muito por desenvolver novas interfaces que permitam o design expedito destas funções.

No que diz respeito às abordagens *Window-Icon-Menu-Pointer* (WIMP), estas tornam a definição e edição destas funções entediante e demorada. Segundo esta abordagem, o design de funções de transferência consiste num processo com várias iterações do tipo tentativa-erro, onde o utilizador empeça com as limitações acima descritas.

Uma forma de colmatar estas limitações consistiria em desenhar directamente da função de transferência e aumentar o retorno de informação sobre a imagem representada para o utilizador, conferindo uma associação de informação espacial à informação espacialmente independente da função de transferência.

O objetivo principal deste trabalho consiste em explorar novas formas de design de funções de transferência para imagens médicas 3D usando uma interface baseada em esboços onde, dando ao utilizador a possibilidade de desenhar livremente a sua própria função directamente sobre uma superfície interativa, se procura diminuir o número de iterações do tipo “tentativa e erro”. Propomos funcionalidades que permitem um rápido reconhecimento e exploração geral do binómio intensidade-anatomia. Uma vez explorados os dados e identificada a estrutura de interesse, o utilizador pode facilmente desenhar a sua função transferência com um menor número de tentativas, logo, num menor intervalo de tempo. Tanto quanto é do conhecimento dos autores, a utilização de esboços e gestos multitoque para o desenho directo de funções de transferência, sobre uma superfície interativa, é um tópico que ainda não foi devidamente explorada.

2. TRABALHOS RELACIONADOS

Existem várias aplicações que permitem a visualização e manipulação de volumes de dados médicos. Exemplos destes software são o OsiriX [OsiriX15], VolView [Volview15], voreen [Voreen15], Exposure Render [Exposure Render15] que permitem a visualização de imagens médicas produzidas por equipamentos de imagiologia médica, assim como a visualização de dados em 2D/3D/4D. Estas aplicações disponibilizam ao utilizador ferramentas para o design da função de transferência unidimensional e bidimensional [Voreen15], aplicação de cores ou a selecção de uma função de transferência dentro de um catálogo de funções previamente definidas.

No entanto, vários investigadores já reconheceram que o desenho de uma função de transferência segundo uma abordagem WIMP é um processo difícil e moroso, surgindo vários trabalhos para mitigar estas limitações [Selver07; Peng10; Chu10]. No entanto, estes trabalhos têm-se focado na criação ou adaptação automática de funções de transferência, que são geradas com base nas características presentes nos dados [Peng10; Chu10]. Alternativamente, surgem interfaces adaptadas ao

processo de desenho iterativo das funções, mas sem alterar a forma de como a função é definida [Selver07].

3. METODOLOGIA

3.1 Representação Tridimensional

A representação de volumes aqui usada consiste em usar um sistema de partículas onde cada partícula representa a cor e opacidade de um determinado voxel [Kroes12; Voglreiter12]. Este tipo de representação gráfica é adequada para estudos mais focados em interacção do que em visualização, pois os sistemas de partículas têm por norma uma resolução espacial menor e apresentam o artefacto de discontinuidade (i.e., é visível a separação entre partículas).

3.2 Função de Transferência

A função de transferência consiste num mapeamento dos valores originais de intensidade do voxel em propriedades ópticas como cor e opacidade. Estas funções podem ser classificadas como sendo unidimensionais (apenas aceitam as intensidades dos voxels) ou bidimensionais (aceitam a intensidade e norma do gradiente da intensidade) [Arens10]. Também podem ser classificadas como sendo espaciais (caso um dos valores do domínio seja uma variável espacial) ou não espacial (caso do histograma unidimensional). Neste trabalho, consideramos apenas funções de transferência unidimensionais.

Graficamente, a função de transferência faz-se acompanhar pelo histograma 1D do volume, geralmente apresentando-se em segundo plano na figura. O histograma 1D é uma representação gráfica da distribuição do número de voxels presentes no volume que possuem um dado valor discretizado de intensidade, sendo o domínio limitado pelo valor máximo e mínimo de intensidade dos voxels presentes nos dados. Assim sendo, o design de funções de transferência é muito influenciado pela informação proveniente do histograma.

3.3 Sistema Interactivo para o Desenho de Funções de Transferência

O sistema é constituída por uma mesa de grande formato, televisão de grande escala, disposta na horizontal, onde é feita a visualização dos dados, e um sistema sensores multitoque sobre a mesa, responsável pela deteção do contacto do utilizador com a superfície (Figura 1). A aplicação foi desenvolvida utilizando o motor Unity3D [Unity3D15] e executada no sistema operativo Windows 7. O computador utilizado possui um processador Intel® Core™ i7-3770K, 3.50 GHz, 16 GB de memória RAM e uma placa gráfica NVIDIA GeForce GTX 970, com 3906 MB de VRAM.

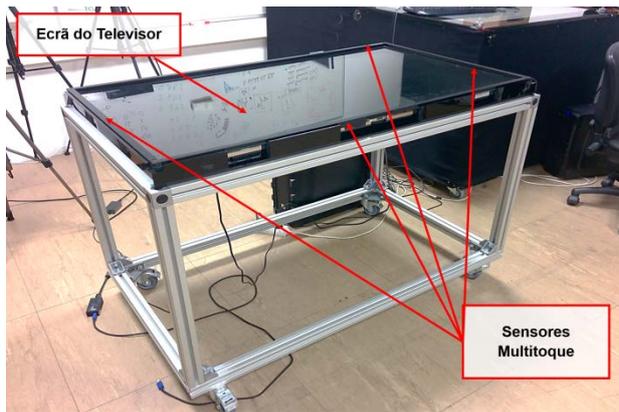


Figura 1 - Equipamento da interface com a aplicação

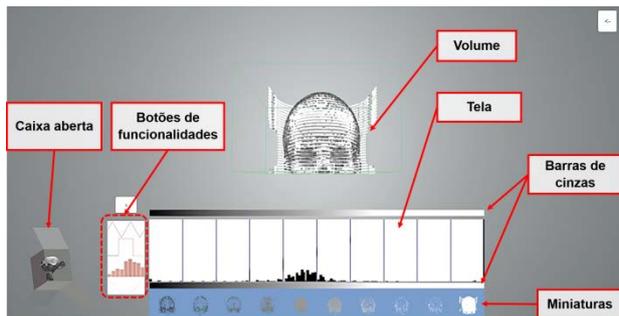


Figura 2 - Interface gráfica do sistema interativo

Em termos de interface gráfica (Figura 2), a aplicação disponibiliza o volume de imagens médicas na zona central superior do ecrã, sendo a zona inferior dedicada às ferramentas de design. A zona de design é constituída por uma tela branca, dividida em dez divisões, sobre a qual serão dispostas e desenhadas as funções de transferência. Existem também duas barras de tons de cinza acima e abaixo da mesma, bem como um conjunto de miniaturas no fundo da tela e uma caixa aberta no lado esquerdo do ecrã, cujas funções serão descritas nas subsecções que se seguem. Existe também um conjunto de botões diretamente à esquerda da tela que permitem a selecção das várias funcionalidades de esboço que pretendemos explorar. Estes botões são para as funcionalidades rampa, janela, régua, esboço, respectivamente.

3.4 Design das Funções de Transferência

Foram desenvolvidas funcionalidades inovadoras para o esboço e design de funções de transferência usando uma tela branca retangular. Esta tela apresenta em segundo plano o histograma 1D das intensidades dos voxels. Central às formas de design propostos, definimos uma divisão equidistante dos domínios da função de transferência e histograma em 10 segmentos, cada um representando 10% do domínio. Esta divisão serve, essencialmente, para ajudar o utilizador a interpretar os dados do histograma e, conseqüentemente, para auxiliar o design da função de transferência.

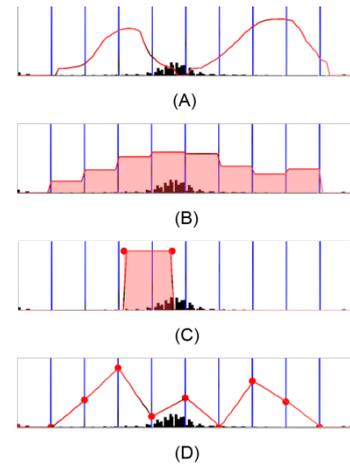


Figura 3 - Técnicas de design de funções de transferência: (A) esboço, (B) régua, (C) janela deslizante, e (D) função rampa.

O design pode ser separado em duas categorias: (i) guiado por dados (informação derivada do conjunto de dados); e (ii) guiado por imagens (utiliza as imagens do volume apresentado) [Kindlman02]. No caso da interface proposta, estamos perante um processo de desenho da função de transferência guiado por imagens. Ao todo, consideramos 5 tipos de técnicas para o design da função de transferência inicial como estão descritas nas subsecções que se seguem (Figura 3).

3.4.1 Miniaturas

Com o objetivo de facultar uma descrição global do volume de dados, é apresentado um conjunto de miniaturas disposto debaixo da tela. Cada miniatura apresenta os voxels representados pelos valores do respectivo intervalo do domínio (Figura 4).

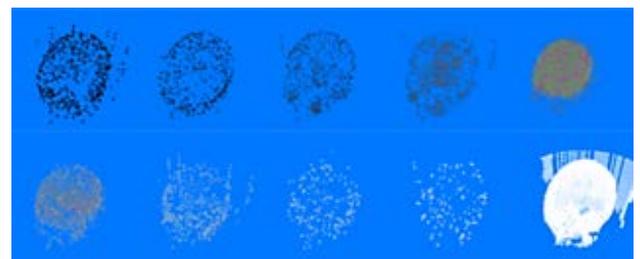


Figura 4 - Exemplo do uso de miniaturas para um volume de Tomografia Computorizada Cerebral.

Assim, cada miniatura representa 10% do domínio do histograma, facilitando a interpretação dos dados do histograma. Esta funcionalidade serve também para compensar a falta de informação espacial associada à função de transferência unidimensional, criando uma relação direta e visual entre os dados da função, nomeadamente a intensidade dos voxels, e as estruturas do volume.

3.4.2 Esboço

É dada ao utilizador a opção de desenhar livremente qualquer esboço de uma função de transferência sobre a tela (Figura 3 (A)). Uma vez desenhada a função, este é aplicada ao volume em tempo-real, vindo a substituir a função anterior. A função pode ser então modificada de

duas formas: pode ser adaptada ou redesenhada. Se um traço pequeno (menos de 50% do comprimento da tela) for desenhado, então a função ao longo do comprimento do traço é substituído por um novo segmento, adaptado do traço desenhado. Isto permite ao utilizador fazer alterações mais precisas na função de transferência, sem a necessidade de redesenhar a função na sua íntegra. No entanto, se o traço for superior a 50% do comprimento da tela, toda a função é substituída por uma função nova, adaptada ao esboço novo.

3.4.3 Réguas

Cada um dos 10 segmentos do domínio da função de transferência é sobreposto por um botão deslizante semi-transparente (Figura 3 (B)). Cada botão deslizante pode ser arrastado ou posicionado a qualquer altura dentro da tela, sendo que a posição do limite superior do botão representa a opacidade a ser atribuída aos valores da função dentro do intervalo desse respetivo botão. Vários botões podem ser movidos simultaneamente, constituindo esta técnica numa metáfora de um DJ numa mesa de som (Figura 5).

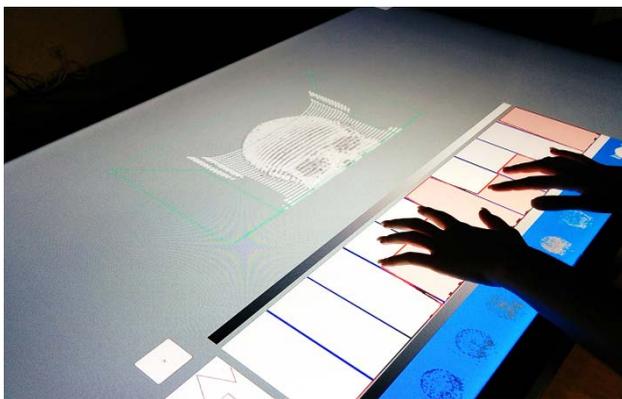


Figura 5 – Exemplo do uso de réguas seguindo a metáfora de “DJing” para obter uma aproximação inicial da função de transferência.

Note-se que as próprias miniaturas podem ser usadas como botões on/off, reduzindo a opacidade do intervalo a transparente ou opaco.

3.4.4 Janela

A janela deslizante pode ser usada para explorar todo o domínio da função de transferência, destacando os pontos representados pelo domínio contido no intervalo da janela (Figura 3 (C)). Esta janela comporta-se como um sinal quadrado, com os valores do domínio da função contido entre os limites da janela tendo um valor de opacidade proporcional à altura da mesma e os pontos fora do intervalo possuindo transparência total. A altura e largura da janela pode ser facilmente ajustada arrastando dois cursores nos seus limites superiores, sendo que a altura e largura máxima estão limitadas às dimensões da tela. A janela pode ser arrastada sobre todo o domínio, permitindo assim o varrimento dos dados na íntegra, com uma resolução definida pelo comprimento da janela utilizada.

3.4.5 Rampa

A função rampa consiste num conjunto de segmentos de recta unidos entre si por nós (Figura 3 (D)). É este tipo de

função que tipicamente se encontra nos softwares de visualização de dados volúmicos [Osirix15; VolView15; Voreen15; Exposure Render15]. A função assim definida pode ser editada através do movimento dos nós ao longo do espaço da tela, sendo que os valores intermédios entre dois nós são determinados por interpolação linear, à semelhança das aplicações tradicionais.

3.5 Protocolo para o Design de Funções de Transferência

Cada técnica aqui apresentada para o design de funções de transferência tem uma forma específica de ser utilizada, podendo ter como objetivo obter uma aproximação inicial da função de transferência ou o refinamento da mesma. As funcionalidades da janela, rampa e régua permitem uma aproximação inicial da função de transferência. No entanto, estas funcionalidades não permitem um refinamento da função, sendo que este passo terá de ser levado a cabo pela função de esboço, que permite um maior controlo de edição da função.

Quando o utilizador troca de funcionalidades, a função de transferência prévia é adaptada à nova funcionalidade, retendo parte da sua informação. A informação perdida está associada à diferente forma como as funcionalidades afectam a função de transferência. Por exemplo, a funcionalidade esboço mantém a função inalterada enquanto que as réguas fazem a média dos valores da função no intervalo de cada régua para determinar a sua posição.

Ao manipular volumes com funções de transferência, a perda de informação dificulta a análise dos volumes, aumentando o tempo necessário para obter os resultados pretendidos, contrariando os objetivos do projeto. Desta forma, torna-se importante não só determinar quais as funcionalidades a utilizar, mas também em que ordem devem ser utilizadas.

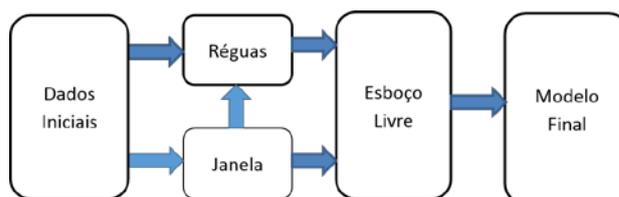


Figura 6 - Esquema ilustrativo da ordem de utilização das funções de desenho da função de transferência

Assim, protocolamos que a manipulação das representações deva ser feita de forma a que nunca se transite de uma funcionalidade com maior informação para uma de menor informação, resultando numa sucessão de técnicas representada pela (Figura 6).

3.6 Manipulação Geométrica do Volume

A visualização do volume exige a definição de pontos de vista. Para tal, a manipulação do volume de dados é feita exclusivamente por gestos multitóque. Arrastando o dedo pela superfície de toque, o utilizador pode rodar o volume em qualquer direção, sendo que a rotação irá acompanhar o movimento do dedo. Utilizando dois dedos, o utilizador é capaz de fazer zoom in ou zoom out (afastando ou aproximando os dedos, respectivamente) e de rodar o

volume segundo o eixo vertical definido pelo dedo de primeiro contacto. Se o utilizador tocar com os cinco dedos da mão, então o volume irá acompanhar a posição da mão, sendo possível assim arrastar o mesmo para qualquer ponto do espaço de trabalho. Também é possível devolve-lo à sua posição e rotação inicial com três toques consecutivos na superfície de toque.

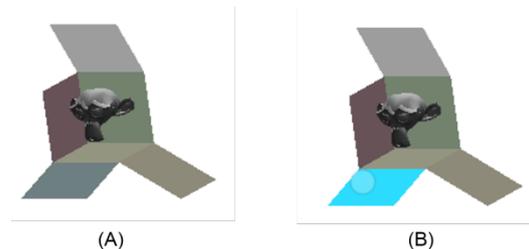


Figura 7 – Grafismo usado para a caixa aberta em modo (A) faces desactivas e (B) face (anterior) activa.

Para auxiliar a manipulação, existe também uma caixa aberta (Figura 7) no canto inferior esquerdo que permite a visualização do volume segundo a orientação das faces do cubo, como por exemplo, ao seleccionar a face inferior do cubo, a face inferior do volume seria visualizada.

3.7 Atribuição de Cores

Quando é feita a análise de imagens médicas em 3D, a atribuição de cores é importante pois permite a distinção de tecidos semelhantes (leia-se, associados a valores de intensidade) quando estes possuem opacidades iguais ou indistinguíveis ao utilizador.

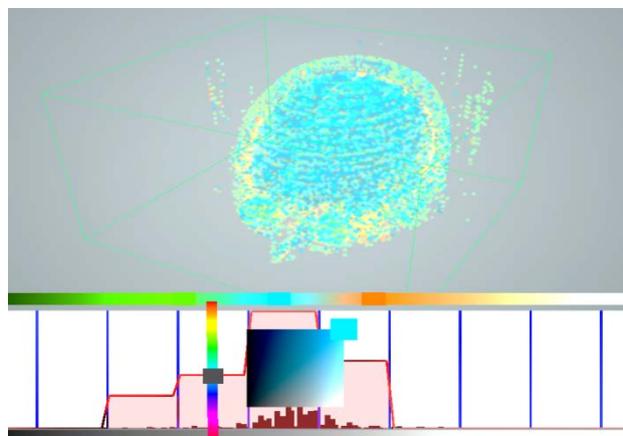


Figura 8 - Exemplo de aplicação dos cursores de cor, possuindo três cursores, interpolação de cores entre eles e uma tela de cores aberta abaixo do terceiro cursor.

Neste artigo propomos uma atribuição de cores simples, com uma ocupação do ecrã mínima e que tire partido da interacção por gestos (Figura 8). Para alcançar isto utilizamos as duas barras de escalas de cinzento, posicionadas diretamente acima e abaixo da tela. A atribuição de cores é feita interagindo com a barra superior: um duplo toque nesta barra vai criar um pequeno cursor que possui uma dada cor. Premindo o cursor uma única vez abre ou fecha uma tela de selecção de cores, que consiste numa barra de cores e de um quadrado que apresenta as diferentes saturações e luminosidades da mesma. A selecção da cor é aplicada ao

cursor em tempo real e a cor é atribuída ao volume. A posição do cursor na barra vai indicar o valor do domínio a que a cor será aplicada, sendo então interpolada até aos limites da barra ou até a outra cor, caso estejam presentes outros cursores (Figura 8).

A eliminação de cores é feita arrastando os cursores para o exterior da barra, largando-os simplesmente fora da área da tela. Se o cursor for largado noutra ponto em cima da barra, este simplesmente mudará de posição. Caso seja largado dentro da tela, dois cursores serão criados e posicionados nos extremos do intervalo em que o cursor foi largado, para que o mesmo tenha essa cor.

4. TESTES DE UTILIZADORES

Foi realizado um conjunto de testes com utilizadores do sistema desenvolvido, com o intuito de obter informação sobre as mais-valias e limitações que a mesma possa possuir no design de funções de transferência.

Os testes de utilizador realizados foram levados a cabo com três estruturas anatómicas, em que para cada estrutura era apresentada uma imagem do resultado pretendido e o utilizador teria de usar as ferramentas para obter um resultado semelhante (Figura 9).

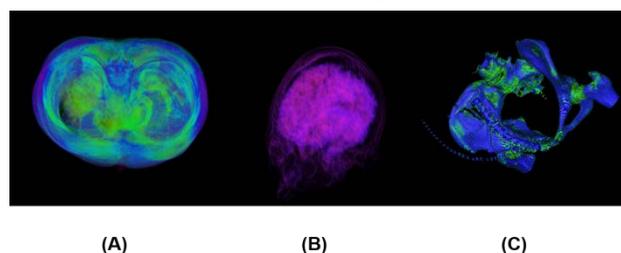


Figura 9 - Imagens de teste: (A) Angiograma Renal, (B) Ressonância Magnética Cerebral, e (C) Tomografia Computorizada da Anca.

As imagens utilizadas advêm de um angiograma renal, uma ressonância magnética cerebral e uma tomografia computadorizada da anca (Figura 9), denominados como testes 1, 2 e 3 respectivamente. Estas imagens foram obtidas através da base de dados de imagens do OsirX, Denominadas como BEAUFIX, BRAINIX e PELVIX, respectivamente [OsiriX15].

Os testes foram realizados com 9 pessoas, com idades de 24 anos \pm 5 anos, todas pertencentes a cursos de Engenharia (Licenciatura ou Mestrado). Destas pessoas, sete eram homens e duas eram mulheres. Todos os utilizadores possuíam dispositivos com controlos de toque e utilizavam os mesmos várias vezes ao dia. Antes dos testes, sete dos utilizadores já possuíam alguma experiência ou noção básica de imagens médicas, mas dois dos utilizadores não possuíam nenhuma experiência.

Os testes foram realizados em duas plataformas: No software de visualização de imagens médicas VolView [VolView15], e na plataforma desenvolvida por nós. Seguidamente foram preenchidos questionários sobre as experiências do utilizador, para serem posteriormente analisados.

Foram cronometradas os tempos de execução dos três testes para cada sistema (Tabela 1). Verifica-se uma

melhoria significativa para os 2 primeiros testes quando os utilizadores recorreram às novas ferramentas, sendo no último teste os tempos são bastante similares (Figura 10). Os tempos contemplam a aplicação da janela deslizante, uso de réguas, esboço, aplicação de transformações geométricas sobre o volume e selecção de cores.

	Teste 1	Teste 2	Teste 3
VolView	$\mu = 7,99$	$\mu = 8,05$	$\mu = 3,90$
	$\sigma = 2,24$	$\sigma = 2,55$	$\sigma = 0,89$
Sistema Interactivo	$\mu = 6,15$	$\mu = 4,89$	$\mu = 4,71$
	$\sigma = 2,51$	$\sigma = 2,84$	$\sigma = 1,57$

Tabela 1 - Tempos de execução das tarefas para cada um dos sistemas usados (em minutos).

Após os testes, a maioria dos utilizadores preferiram os comandos e ferramentas da nossa plataforma (8 dos 9 utilizadores preferiram as ferramentas apresentadas na plataforma), classificando estes como sendo de fácil utilização, comparativamente às ferramentas tradicionais (7 dos 9 utilizadores consideraram as plataforma de mais fácil utilização no geral). A maioria (7 dos 9 utilizadores) também preferiu a utilização de gestos para o desenho das funções e utilização das ferramentas.

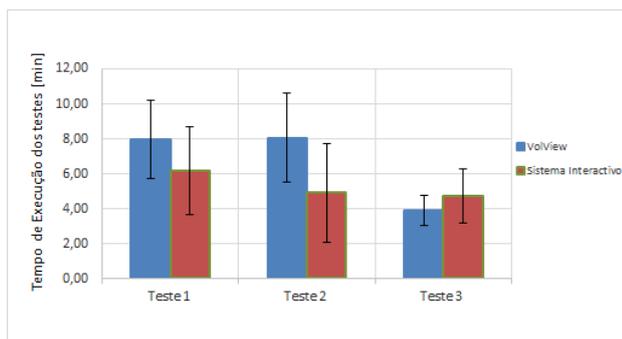


Figura 10 - Tempos de execução das tarefas (em minutos).

De entre as ferramentas apresentadas, a funcionalidade das réguas foi a mais utilizada e apreciada por parte dos utilizadores para obter uma aproximação da função de transferência (7 dos 9 utilizadores referiram-na como preferida para obter a aproximação das funções), sendo que alguns utilizadores dispensaram a utilização da janela e começaram a aproximação directamente com as réguas. Um feedback notável foi o facto da grande maioria dos utilizadores (8 dos 9) considerar que a presença das miniaturas constituía uma mais-valia na compreensão dos dados.

Também foi registada alguma preferência pela simplicidade da tela de cores utilizada na nossa plataforma. Igualmente, foi registada uma preferência dos controlos de toque para a manipulação na posição e rotação do volume (8 dos 9 utilizadores afirmaram tal), sendo a caixa-aberta considerada uma mais-valia (8 dos utilizadores afirmaram que a sua presença era benéfica), apesar de ser pouco utilizada na maioria dos casos.

5. DISCUSSÃO

Neste trabalho, explorámos o design de funções de transferência através de gestos e se as técnicas desenvolvidas permitiam obter, de forma expedita e natural, aproximações iniciais de funções de transferência, assim como se permitem uma compreensão mais imediata da distribuição espacial das intensidades em imagens médicas 3D. Desta forma, o sistema desenvolvido visa facilitar o esboço e edição da função de transferência, utilizando funcionalidades que assentam no uso de gestos para a sua utilização.

Os resultados obtidos revelaram-se muito satisfatórios no sentido em que os utilizadores mostraram uma clara preferência pelos métodos baseados em gestos. Quando confrontados com o ambiente de trabalho WIMP, os utilizadores notaram uma falta de informação em relação aos dados do volume apresentado, sendo que isto levou a dificuldades claras na manipulação da função de transferência. Estas dificuldades foram claramente mitigadas com a presença das miniaturas, levando a uma menor quantidade de tentativas até chegar a uma aproximação aceitável do volume.

A utilização de esboços também levou a que os utilizadores fossem capazes de conferir a forma desejada à função de transferência de forma mais fácil. No entanto, a liberdade conferida ao utilizador para o esboço das funções de transferência serviu ocasionalmente como um obstáculo, sendo que alguns utilizadores inadvertidamente realizavam alterações às funções de transferência que produziam resultados indesejados e substanciais no volume. Mesmo em casos extremos, em que a função teria de ser redesenhada de início, os utilizadores foram capazes de obter uma nova aproximação de forma relativamente rápida. Tal já não acontecia com as ferramentas tradicionais, onde os utilizadores tinham mais dificuldades em desfazer resultados indesejados.

Curiosamente, e em ambas as ferramentas, notou-se alguma dificuldade da parte dos utilizadores na atribuição inicial de uma cor. Foi reportado que não parecia existir uma correspondência imediata entre cor seleccionada e a respectiva coloração da estrutura anatómica de interesse. Por outras palavras, a selecção de uma cor não estava devidamente assinalada no volume representado, sendo necessário mover o cursor da cor pelo domínio para se ter noção da coloração aplicada. Ainda assim, os utilizadores revelaram preferência pela selecção e atribuição de cores usando o sistema interactivo pois permite a selecção directa da matriz de cor, saturação e brilho.

Os tempos de execução para desempenhar os testes foram menores usando o sistema proposto e similar no último teste (Tabela 1). Apesar de no geral os utilizadores terem demorado menos tempo na realização das tarefas na nossa plataforma, em várias instâncias terá ocorrido demoras na obtenção de resultados em ambos os casos devido a utilização indevida com as ferramentas ou falta de familiaridade com as mesmas ou os dados a serem utilizados. Para além disso, em alguns casos, os utilizadores ao invés de utilizarem as ferramentas para

realizar a tarefa em si, usaram-nas numa forma mais lúdica, devido à natureza singular das mesmas. Desta forma, uma análise mais detalhada dos tempos necessários para a utilização de cada ferramenta, com objectivos mais bem definidos, seriam uma mais-valia para saber objectivamente o impacto das novas ferramentas.

O estudo realizado mostra que uma interacção por gestos é não só adequada para obter uma aproximação mais fácil da função de transferência, mas também permite uma melhor exploração dos dados, principalmente quando aliada à apresentação de informação *a priori* sobre os dados no domínio da função.

6. CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho propusemos que um conjunto de ferramentas para esboçar uma aproximação inicial de uma função de transferência e diminuir as dificuldades encontradas no desenho das mesmas. Os resultados obtidos mostram uma preferência dos utilizadores por controlos baseados em gestos, tanto no esboço das funções como no controlo dos volumes, e uma resposta positiva à apresentação de informação *a priori* sobre os dados, bem como a sua integração directa com as ferramentas.

Como trabalhos futuros, será necessário conduzir mais testes com utilizadores para atingir maior significância estatística. Adicionalmente, um novo conjunto de testes será realizado de forma a analisar em maior detalhe de que forma, e em que tarefas, as novas funcionalidades têm uma contribuição para a obtenção de melhores resultados. Estes testes serão mais focados numa análise mais pormenorizada de cada tarefa e como cada uma é afectada pelas ferramentas de ambas as plataformas.

Seguidamente, explorar novas formas de associar as modificações dos dados às alterações da função de transferência, de forma a tornar mais imediata a compreensão dos dados ao utilizador.

Futuramente, procura-se também a integração destas ferramentas com funções de transferência bidimensionais, sendo que estas possuem mais informação espacial, tornando-as capazes de uma melhor separação de estruturas anatómicas que possam possuir valores semelhantes de intensidade. [Arens81]

7. AGRADECIMENTOS

Os autores agradecem o apoio financeiro concedido pela Fundação para a Ciência e a Tecnologia, nomeadamente pela bolsa de pós-doutoramento com a referência SFRH/BPD/97449/2013 e pelos fundos nacionais com a referência UID/CEC/50021/2013 e TECTON-3D PTDC/EEI-SII/3154/2012.

8. REFERÊNCIAS

- [Arens81] Arens S., Domik, G., A Survey of Transfer Functions suitable for Volume Rendering, IEEE/EG International Symposium on Volume Graphics 0, 0, 1–5 (1981).
- [Bergeest08] Bergeest, J.-P. Bergeest, Jäger, F., A Comparison of Five Methods for Signal Intensity Standardization in MRI Bildverarbeitung für die Medizin 2008, Bild. für die Medizin 2008 SE - Inform. aktuell, 36–40, (2008).
- [Botha12] Botha, C, Preim, B., Kaufman, A., Takahashi, Ynnerman, A. From individual to population: Challenges in Medical Visualization, Scientific Visualization Mathematics and Visualization, Chapter 23, 265–282, (2014).
- [Chu10] Chu, Haiyang, Chen, Li, Yong, Junhai, Feature variation curve guided transfer function design for 3D medical image visualization, Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference, 1, 0, 257–263, (2010)
- [Exposure Render15] Exposure Render
<<https://code.google.com/p/exposure-render/>>
- [Goodman09] Goodman, A., Seeing Science, Simbios, an NIH National Center for Biomedical Computing, 28, 4, (2009)
- [Kindlmann02] Kindlmann, G., Transfer functions in direct volume rendering, ACM SIGGRAPH, Course Image Process., Graph., d, (2002)
- [Kroes12] Kroes, T., Post, F.H., Botha, C.P., Exposure Render: An Interactive Photo-Realistic Volume Rendering Framework, PLoS One, 7, (2012)
- [Lundström06] Lundström, C., Ljung, P., Ynnerman, A., Local histograms for design of transfer functions in direct volume rendering, IEEE Trans. Vis. Comput. Graph., 12., 6, 1570–1579, (2006)
- [McGhee10] McGhee, J., 3-D visualization and animation technologies in anatomical imaging, J. Anat., 216, 2, 264–270, (2010).
- [OsiriX15] OsiriX Imaging Software
<<http://www.osirix-viewer.com/AboutOsiriX.html>>
- [Peng10] Peng, Yi, Chen, Li, Multi-feature based transfer function design for 3D medical image visualization, Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference, 1, 0, 410–413, (2010)
- [Selver07] Selver, M. Alper, Fischer, F., Kuntalp, M., and Hillen, W., A software tool for interactive generation, representation, and systematical storage of transfer functions for 3D medical images, Comput. Methods Programs Biomed., 86, 3, 270–280, (2007)
- [Unity15] Unity Technologies
<<https://unity3d.com>>
- [VolView15] VolView
<<http://www.kitware.com/opensource/volview.html>>
- [Vogltreiter12] Vogltreiter, P., Stochastic Particle-Based Volume Rendering, Proceedings of CESC2012: The 16th Central European Seminar on Computer Graphics, (2012)
- [Voreen15] Volume Rendering Engine
<<http://www.voreen.org/>>

Enabling low-complexity devices for interaction with 3D media content via Android API

Ricardo Santos¹
¹Polytechnic Institute of Leiria / ESTG
 Leiria, Portugal
 ricardo.a.santos@ipleiria.pt

Hugo Costelha^{1,2}, Luis Bento^{1,3}
 and Pedro Assunção^{1,4}
²INESC TEC, ³ISR-UC,
⁴Instituto de Telecomunicacoes
 Leiria, Portugal
 {hugo.costelha, luis.conde,
 pedro.assuncao}@ipleiria.pt

Márcio Barata⁵
⁵Tech4Home
 São João da Madeira, Portugal
 marcio.barata@tech4home.pt

Abstract

This paper deals with an interactive multimedia system based on Android OS, where several functional modules were developed to enable the use of low-complexity remote control devices. The system architecture comprises the remote control device with Magnetic, Angular Rate, Gravity (MARG) sensors for 3D motion tracking and an Android set-top-box, integrating a novel Application Programming Interface (API), specifically developed for this purpose. A proper decision whether the most complex functions should run on the remote control device, or on the Android set-top-box, is an open issue and depends on the specific application and the desired portability. Therefore taking into account energy consumption when balancing the computational burden is paramount. Given that the set-top-box has no limitations on energy consumption and has a superior computational power, the propose API can perform all the processing of sensors data, allowing the implementation of complex fusion algorithms with higher precision. The analysis of energy consumption on the remote control device shows that transmitting the raw sensors data, to be processed in the API, results in lower energy consumption in the remote control device, and consequently higher autonomy with good accuracy.

Keywords

Android, API, Set-Top-Box, USB, HID

1. INTRODUCTION

In the past years there has been a strong investment in technology development for television and multimedia consumer market in general. Besides the evolution of screen resolutions, there has been an evolution that is bringing new types of multimedia content. Before this evolution, the user had a limited interaction with the available content in the television, but the trend is to have more interactive multimedia content and applications. However the devices used for interaction did not follow this evolution, leading to a poor Quality of Experience (QoE). The Remote Control Device (RCD) of current Set-Top-Box (STB) or smart TV is used for interaction with multimedia content, mainly based on two dimensions (2D) [Ohnishi 12]. The evolution to 3D content and operation with enhanced interactive functionalities, requires mapping of the RCD movements into motion on the 2D screen [Zidek 13]. To reduce integration barriers, the main manufacturers are moving towards Android-based systems. This operating system has increasingly been adopted for multimedia services both on television¹ and STB¹ [Song 10]. Since Android is an open system, it provides easy access to its internal architecture,

allowing faster development, implementation and testing of Application Programming Interfaces (API).

In the scope of this research, a system for 3D interaction with multimedia content was developed and tested. This system, comprised of an RCD transmitting sensors data to an STB, includes three functional modules: communication, processing and application layer. One of the challenges addressed in this work is related to compute orientation estimates in the RCD. This is because running complex algorithms in the RCD, results in higher energy consumption for processing but lower energy consumption for communications, due to less data being transmitted. Therefore, a proper computational balance between the RCD and the STB is important, *i.e.* one has to decide whether the most complex functions (in terms of computational complexity) should run in the RCD or in the STB, taking into account energy consumption. Transmitting RAW data to the STB increases the energy required for communications, but allows the implementation of more complex algorithms on the STB, thus leading to more accurate estimates.

¹<https://www.android.com/tv/>

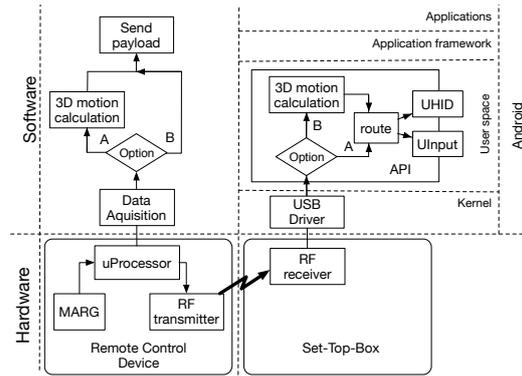


Figure 1. System architecture.

The Application Programming Interface (API) was designed to have the least possible impact on the Operating System (OS), i.e., the API does not require any change in the Android OS kernel and frameworks, therefore simplifying its use. The API receives data from RCD and makes it available to the OS after the computation process. The implementation on the STB side, allows access to information about the user system (e.g., available resources).

This article addresses the analysis of the computational balance between the RCD and the STB. The sensors data processing – implementing an Air mouse that behaves like a pointer – can be done in the RCD or in the STB, depending on the evaluation of both the energy consumption profile and QoE [Rasteiro 15].

2. SYSTEM ARCHITECTURE

The system architecture is comprised of two functional entities: hardware and software. Figure 1 presents the overall system architecture. The system hardware consists of an RCD with 6 Degrees of Freedom (DoF) and a STB. The RCD is able to track 3D motion using a set of sensors commonly known as Magnetic, Angular Rate and Gravity (MARG). MARG Sensors are composed by an accelerometer, gyroscope and magnetometer, each with 3 orthogonal axes. In this research we have used a MARG that also includes an Application-Specific Integrated Circuit (ASIC) embedded processor, designated as Digital Motion Processor (DMP), which computes the orientation of the device using the information retrieved by the accelerometer and gyroscope sensors. The Android-based STB runs the OS 5.0. The STB has no limitations on energy consumption and a far superior computational power, ROM and RAM memory, when compared to the RCD.

To evaluate the problem of computational load balance, two solutions, identified as option “A” and “B” (see figure 1), were implemented:

Option “A” RCD movement is acquired by the MARG unit, the raw data is used by the RCD processing unit, and the processed data is transmitted to the STB;

Option “B” All raw data is transmitted from the RCD to the STB where it is processed.

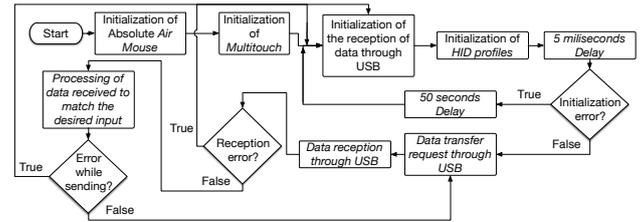


Figure 2. API flowchart.

The processed data in Option “A” results in far less data to be transmitted, as described later.

An external module (USB dongle) was developed to implement the communication between the RCD and the STB via Radio Frequency (RF). The dongle receives data from the RCD through ZigBee[®] Radio Frequency for Consumer Electronics (RF4CE) protocol and forwards them to the STB through Universal Serial Bus (USB) 2.0 Human Interface Device (HID) custom [USB-IF 00], [USB-IF 01].

3 SET-TOP-BOX

Although the STB is based on Android OS and the communication is made through an USB dongle, it is not possible to use the HID software stack to implement all the interfaces (e.g., absolute mouse HID) because the data must be pre-processed before making it available to the OS. Another limitation imposed by the OS, is that it does not allow to get the mouse pointer coordinate values when absolute coordinates are used. The system was implemented through an API running in STB for both options (“A” and “B”), that was designed to have the least possible impact in the STB, to avoid the need for recompiling the kernel or the Android OS.

The API was implemented in the Hardware Abstraction Layer (HAL) user space, because this layer is not hardware-dependent and allows receiving data from any communication interface (e.g. Bluetooth, I2C, SPI, etc.), as shown in figure 1.

The main purpose of the API is:

Option “A” to integrate the processed sensors data from RCD with screen information from the user’s setup.

Option “B” to perform all the heavy processing that requires a great deal of power consumption in the RCD and integrate the result data with screen information from the user’s setup.

The API was developed in the C programming language as a native application. For the development of such applications there is a Native Development Kit (NDK) available for Android devices. However this was not used, since it requires an application in Java for Android to run and start the native application. Thus, only the GNU Compiler Collection (GCC), available with NDK, was used to build the native application developed for the Android device.

The application flowchart can be observed in figure 2. The native application must start when the device is powered, therefore one had to make changes to the bootloader, to

```

1 ...
2 service hermes /data/local/API/REMOTE6DOF
3   class main
4   user root
5 ...

```

Code 1. `init.< device >.rc`

load the API as a service. Those changes consisted in editing the “`init.< device >.rc`” file (code shown in code 1), followed by the build and flash of the boot image into the device [Yahmour 13].

Given that the API is in the HAL level, it is possible to declare HID profiles and inputs that make the data available for the entire OS and respective applications at higher layers. Figure 3 shows the interaction between the blocks of the API in the Android software stack. After starting the API, two input profiles on the Android OS are declared and initialised: the pen and multitouch inputs. These profiles were created using a module in user space to create and handle the input devices, *i.e.* an “`uinput`” kernel module. To create a new virtual device the following sequential actions have to be taken:

- 1 Open the user interface (“`/dev/uinput`”) and create a temporary device;
- 2 Publish which input events the device will generate;
- 3 Create a structure with the basic information of device, namely the maximum and minimum values for the input events;
- 4 Send the command to the interface to create the device.

The pen input device was implemented to overcome the Android OS limitation of not making the pointer visually available for the absolute mouse. Two conditions have to be fulfilled in order to make the pointer visible on the screen: (i) explicit configuration of the requirement for a pointer and (ii) claim that the pen is in the range of the screen.

These virtual input devices are initialized taking into account the screen size of the device where it is running. This is done by reading the resolution field “`FBIOGET_VSCREENINFO`” of the framebuffer “`/dev/graphics/fb0`”. The relative mouse, joystick, gamepad and consumer electronic virtual USB HID devices, are created through a similar procedure but this time, using an USB interface in user space “`/dev/uhid`”.

Using the “`libusb`”¹ the USB is started as host in order to receive data from the dongle. If there is an error during any of the initializations, the API waits 50 seconds and tries again, repeating the process until there is no error. This ensures that the API only continues after establishing a proper connection with the dongle. It should be noted that the device may not be connected when the API is started, so through this cycle, it can be ensured that the device is detected with a maximum delay of 50 seconds from the connection.

¹<http://libusb.info>

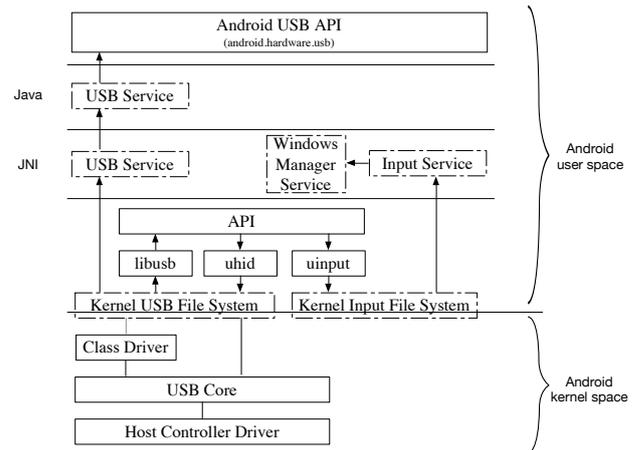


Figure 3. Android API Stack [Regupathy 14].

The initialisation request for information is sent to the dongle and the data is read from the USB buffer. The received data passes through an error checking routine and, if there are no data errors, the data is handled and sent to the corresponding USB HID profile. If an error occurs while receiving or sending data, a soft reset is performed by software, leading to an API reinitialisation which ensures that there is no accumulation of errors.

4 ENERGY CONSUMPTION ANALYSIS ON THE REMOTE CONTROL

The RCD modules consists of a Microcontroller Unit (MCU), MARG sensors and a RF module. In order to chose the best option (Option “A” or “B”) in terms of greater battery life, an analysis on the energy consumption of each module was made. Consumption measurements were taken with the Analog Discovery board that provides a 100 kHz sampling rate and 14 bit resolution.

Five setups were implemented and tested, comprising representative sensors data acquisition and software algorithm implementation. Setups 1, 2 and 3 correspond to the processing in the RCD (Option “A”). Setups 4 and 5 have the computational load in the STB (Option “B”). All setups are presented in table 1.

4.1 Tests Characterization

TheQoE was taken into account in the power consumption tests, since it is affected by the pointer position refresh rate on the screen, which depends on the sensors data processing and transmission frequency. For all tested setups, the data acquisition is done at 100Hz and transmission at 50Hz. In the first setup, the data from the sensors is acquired in RAW and, using the MhF [Rasteiro 15], is computed the device orientation to determine the HID relative mouse position. Results are fitted in 2 bytes (X and Y) and sent via RF, with 1 byte representing the header. The second setup also uses the MhF to determine the HID absolute mouse position, in this setup the results are fitted in 4 bytes (X and Y). The two added bytes arise by matching the range of values with the size of the screen, which requires at least 2 bytes for each dimension. In the setup 3 the device orientation is computed by the DMP, so it

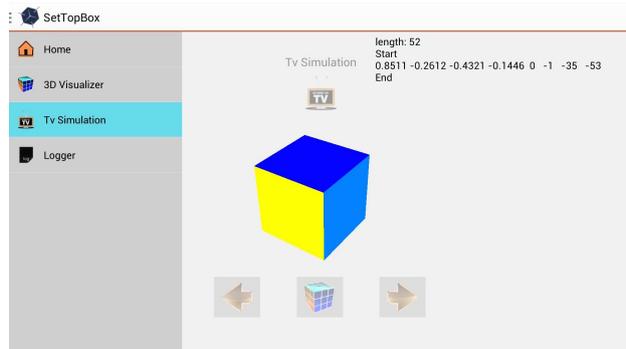


Figure 4. Android application.

is not necessary to apply further data processing. The results are also fitted in 4 bytes (X and Y) has in the previous setup. Setup 4 consists in the acquisition of raw data and its respective transmission, i.e., 3 bytes for each sensor (Accelerometer, Gyroscope and Magnetometer), plus 1 byte for the header. In the setup 5 both DMP and RAW data are acquired, but only the orientation computed by the DMP and the Magnetometer data are transmitted. It consist on 3 bytes for each component of the DMP, 2 bytes for the Magnetometer data and 1 byte for the header.

4.2 Analysis of Results

Setups 1 and 2 were used to test the sensors data processing in the RCD using the RAW data, which results in the relative and absolute mouse, respectively. As expected, the absolute mouse implementation requires more energy, as it needs more processing and more data to be sent. Setup 3 implements part of the sensors data processing in the sensors module through the DMP. Results show that it requires more energy than Setup 2, which performs data processing from the RAW sensors data in the microcontroller. In Setup, 4 RAW data is obtained from the sensors without any processing being done in the RCD. Although this results in more data to be transmitted, this Setup revealed to consume less energy than all the Setups presented above. Setup 5 uses the DMP processing in sensors module in order to transmit less data, however it consumes more energy than the Setup 4. The results listed in table 1 show that there is less energy consumption on the acquisition of RAW data. Setup 4 has the lowest consumption, since no processing is done in the remote, as it consists in reading RAW data and its respective RF transmission. When the processing is performed in the RCD, acquire raw data and processing Mahony filter evidence a lower power consumption compared to acquire DMP plus RAW data from sensors in order to avoid Mahony filter processing.

The QoE was tested using an Android application, shown in Figure 4, specifically developed to simulate and test a potential usage environment. The results of the subjective tests, that were carried out to evaluate how friendly is the RCD to non-expert users, revealed that absolute orientation computed in the STB presents smoother motion tracking and good user experience.

Setup	Data acquisition	Computational processing	Payload [Bytes]	Energy consumption [mJ]	Peak duration [ms]
1	Gyro + Acc + Mag	MhF + Air Mouse Rel.	3	1.1612726	16.69
2	Gyro + Acc + Mag	MhF + Air Mouse Abs.	5	1.2275447	16.92
3	DMPquat + Gyro + Acc + Mag	Air Mouse Abs.	5	1.5198403	17.92
4	Gyro + Acc + Mag	-	19	1.1252908	16.98
5	DMPquat + Gyro + Acc + Mag	-	15	1.4749114	18.18

Table 1. Remote control setups and results.

5 CONCLUSIONS

In this research, the energy consumption analysis revealed that the computational processing of RCD sensors data should be made in the STB. Although more data needs to be transmitted when using this option, less energy is consumed in comparison with the case where the computed orientation is read from the DMP or estimated in the RCD. Our solution was to build the RCD interface through an API implemented in the Android-based STB. The kernel source code of the STB was not modified, in order to maximize compatibility with different vendors and to allow easier deployment without any OS changes. The overall system was successfully tested with good user QoE

Acknowledgment

This work is co-financed by European Union, COMPETE, QREN and Fundo Europeu de Desenvolvimento Regional (FEDER), Project HERMES, Co-promotion N^o 34149.

References

- [Ohnishi 12] T. Ohnishi, N. Katzakis, K. Kiyokawa, and H. Takemura. Virtual interaction surface: Decoupling of interaction and view dimensions for flexible indirect 3D interaction. In *3D User Interfaces (3DUI), 2012 IEEE Symposium on*, pages 113–116, March 2012.
- [Rasteiro 15] M. Rasteiro, H. Costelha, L. Bento, and P. Assuncao. Accuracy versus complexity of MARG-based filters for remote control pointing devices. In *Consumer Electronics - Taiwan (ICCE-TW), 2015 IEEE International Conference on*, pages 51–52, June 2015.
- [Regupathy 14] R. Regupathy. *Unboxing Android USB: A hands on approach with real world examples*. Apress, May 2014.
- [Song 10] M. Song, W. Xiong, and X. Fu. Research on Architecture of Multimedia and Its Design Based on Android. In *Internet Technology and Applications, 2010 International Conference on*, pages 1–4, Aug 2010.
- [USB-IF 00] USB-IF. Universal Serial Bus Specification, April 2000.
- [USB-IF 01] USB-IF. Device Class Definition for Human Interface Devices (HID), June 2001.
- [Yaghmour 13] K. Yaghmour. *Embedded Android*. O’Reilly Media, Inc., 2013.
- [Zidek 13] K. Zidek and J. Pitel. Smart 3D pointing device based on MEMS sensor and bluetooth low energy. In *Computational Intelligence in Control and Automation (CICA), 2013 IEEE Symposium on*, pages 207–211, April 2013.

Acesso Multimodal em Dispositivos Móveis a Vídeo Georeferenciado através da Forma, Velocidade e Tempo

Sérgio Serra

Ana Jorge (*)

Teresa Chambel

LaSIGE, Faculdade de Ciências, Universidade de Lisboa

(*) and Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa
Portugal

{sergioserra99,ananunesjorge}@gmail.com, tc@di.fc.ul.pt

Resumo

O vídeo está a tornar-se uma meio cada vez mais capturado, publicado e acedido na web, a partir de diferentes plataformas e dispositivos. Os utilizadores podem mais facilmente georeferenciar a informação que capturam e acedem, permitindo enriquecer a sua contextualização. A procura de vídeo tem sido limitada essencialmente a palavras-chave, ou a um conjunto de parâmetros com suporte muito limitado às dimensões espacial e temporal.

Propomos novos mecanismos para procurar e aceder a vídeos georeferenciados, onde as dimensões espacial e temporal são centrais, permitindo um acesso baseado na forma e velocidade das trajectórias inerentes a estes vídeos, usando uma interface multimodal num dispositivo móvel, baseada em gestos e movimento, e com o potencial de uma interacção mais natural, um maior envolvimento, sensação de presença e imersão. A avaliação com utilizadores, feita com um protótipo de alta fidelidade teve resultados positivos e encorajadores. Os utilizadores consideraram a maioria das funcionalidades úteis, satisfatórias, por vezes divertidas, e fáceis de usar. As diferentes opções e modalidades foram consideradas interessantes e adequadas a diferentes cenários de utilização que foram identificados e sugeridos. Também foram identificados alguns desafios para serem levados em conta em desenvolvimentos futuros, em direcção a um acesso interactivo mais flexível e eficaz, através de uma interacção mais natural com dispositivos móveis de per se, ou como segundos ecrãs em conjunto com ecrãs de maiores dimensões em TV ou ecrãs públicos.

Palavras-chave

Vídeos Georeferenciados, Interação Multimodal, Gestos e Movimento, Procura, Acesso e Navegação, Espaço e Tempo, Forma e Velocidade, Trajectórias, Dispositivos Móveis e Segundo Ecrã.

1. INTRODUÇÃO

Todos os dias, uma grande quantidade de vídeos digitais é publicada na web. O vídeo está a tornar-se amplamente capturado, partilhado e acedido através de diferentes plataformas e dispositivos, que de forma crescente pode ser georeferenciado (em localizações pontuais e ao longo de trajectórias), permitindo enriquecer a sua contextualização. Apesar de muitos vídeos estarem disponíveis para ser procurados e acedidos, os mecanismos actuais e mais usados para os procurar e navegar são baseados num conjunto limitado de parâmetros como: palavras-chave, duração, e qualidade do vídeo, ignorando as suas dimensões temporais e espaciais. O vídeo tem um enorme potencial para imersão e os dispositivos móveis permitem aceder à informação enquanto ‘imersos’ na realidade, em qualquer lugar. Com a proliferação de dispositivos como: *smartphones*, *tablets* e mais recentemente *wearables*, poderíamos tirar partido dos sensores multimodais disponíveis para criar novas formas de encontrar e navegar vídeos georeferenciados através do espaço e do tempo (quer ao longo da sua duração e linha de tempo, quer o tempo da sua captura) e espaço, usando interfaces mais naturais, envolvendo gestos e a forma e a velocidade do

movimento, com o potencial de maior envolvimento, sensação de presença e imersão, ao aceder aos vídeos.

Este trabalho surge no seguimento de um outro trabalho prévio no contexto do Sight Surfers [Noronha12] [Ramalho13a,13b], uma aplicação web interactiva para a partilha, visualização e navegação de vídeos 360º georeferenciados e interactivos, como hipervídeos, incluindo *tours/circuitos* em cidades ou actividades mais radicais, como corridas de kart. Estes vídeos podem ser experienciados de forma mais imersiva, isoladamente, ou sincronizados com um mapa ao longo do percurso. Para permitir um suporte mais completo à dimensão espaciotemporal em vídeos georeferenciados, e com o objectivo de aumentar a imersão associada a uma experiência sensorial aumentada, pretendemos criar mecanismos mais ricos para a procura, visualização e navegação interactivas através de modalidades de interacção mais naturais.

Neste artigo, descrevemos o nosso trabalho desenvolvido nesta direcção. A próxima secção evidencia os principais desafios e oportunidades, e descreve o trabalho relacionado mais relevante. Em seguida, apresenta-se o modelo conceptual e as opções de design para o acesso multi-

modal a vídeo georeferenciado no espaço e no tempo, e dispositivos móveis, que são demonstrados em protótipos, e avaliados na secção seguinte. Foi realizada uma avaliação com utilizadores e protótipos de alta fidelidade, para aferir a usabilidade e o nível de aceitação, focando a utilidade, a satisfação e a facilidade de uso. Finalmente, o artigo termina com conclusões e perspectivas para trabalho futuro.

2. TRABALHO RELACIONADO

Os desafios para este trabalho incluem a concepção e a realização de uma interface interactiva adequada, capaz de capturar e expressar as dimensões temporal e espacial, e permitir a representação da velocidade e da forma das trajectórias ao longo do tempo, através de um modo intuitivo, simples, eficaz e natural de procurar, apresentar e navegar os vídeos em ambientes móveis. É simultaneamente um desafio e uma oportunidade, porque os utilizadores não estão habituados a procurar e a navegar nestas dimensões. Mas a tecnologia já permite capturar o movimento em dispositivos móveis de formas que têm o potencial de suportar modos de interacção mais naturais, envolvendo o tempo, a forma e a velocidade, permitindo prosseguir em direcção a experiências mais imersivas.

2.1 Georeferenciação em Bibliotecas de Vídeo

A maioria das bibliotecas de vídeo e websites como o YouTube ou o Vimeo baseiam-se em palavras-chave e têm quando muito um suporte muito limitado ao acesso a vídeo com base nas dimensões espacial e temporal. Rego et al. [Rego07] desenvolveram a VideoLIB, uma biblioteca digital que melhora a recuperação de vídeo através de operadores espaciais e temporais, baseados nas normas de metadados Dublin Core e MPEG-7. Os critérios de pesquisa incluem: acção (o quê), pessoa (quem), tempo (quando) e local (onde); e usa operadores como: antes, durante e depois para definir intervalos de tempo. Isto permite efectuar pesquisas como: “procura os clips de vídeo da Madonna que foram produzidos fora dos estados unidos da américa durante os anos 1990s”. Utiliza um formulário numa interface baseada em texto, sem o recurso a mapas, e os vídeos são considerados como um todo, e as trajectórias e a velocidade não são tidas em consideração.

2.2 Acesso a Fotos e Vídeos através de Mapas

Há algumas abordagens à procura e navegação de vídeos, e especialmente de fotos, que utilizam mapas. O Google Street View é um visualizador de fotos 360° que usa uma projecção esférica da imagem e geolocalização, mas não suporta vídeo, nem vistas alternativas, inclusivé criadas/capturadas pelos utilizadores, dos diferentes espaços. Panoramio (.com) é um website de partilha de fotos georeferenciadas acedido como um *layer* do Google Earth e Google Maps. Os utilizadores podem efectuar pesquisas baseadas em texto ou navegar nos mapas, e ver as fotos tiradas por outros utilizadores, com base na localização. As fotos são apresentadas juntamente com um mapa que evidencia a sua localização, quer como uma colecção resultante de uma pesquisa, quer de forma individualizada: uma a uma. Existem filtros para evidenciar as fotos mais populares, as mais recentes, locais famosos,

em interiores, que através de um separador próprio para as fotos filtradas quer aumentando a dimensão destas fotos de entre todas as que são apresentadas no mapa.

Finsterwald et al. [Finsterwald 12] desenvolveram o MOMA (The Movie Mashup Application), como um serviço public web baseado em mapas para pesquisar filmes baseados na localização, combinando recursos de georeferenciação e processamento de texto, integrando informação oriunda da DBpedia, GeoNames e Wikipedia. Através da sua interface gráfica, permite procurar e navegar uma base de filmes por realizador, localização por texto, através de áreas poligonais no mapa, a partir de localizações extraídas de títulos de filmes, comparar distribuição das pesquisas e, utilizando uma versão móvel, permite pesquisar filmes em que acção tem lugar numa zona perto da localização actual do utilizador. Apesar de os mapas constituírem uma forma natural de representar informação georeferenciada, e de os vídeos frequentemente envolverem trajectórias, a maioria das soluções existentes só permitem que os utilizadores publiquem e acedam a vídeos com base numa localização GPS única (geralmente a posição inicial). Hao et al. [Hao11] apresentam vídeos gerados pelos utilizadores relacionados com áreas geográficas numa interface baseada num mapa. Eles focam-se na selecção automática de *keyframes* para representar os vídeos, e na determinação da sua localização, para as colocar no mapa. Desta forma, enfatizam os pontos de interesse (exo. um monumento) que são capturados nos vídeos, e não tanto as trajectórias.

2.3 Interacção Háptica para acesso a *Media* em Dispositivos Móveis

Em termos de pesquisa interactiva espacial e háptica em ambientes móveis, nos últimos anos, notou-se uma crescente popularidade das interfaces gestuais e das aplicações em segundo ecrã. Lei & Coulton [Lei09] implementou uma aplicação controlada por gestos que age como uma varinha, usando sensores móveis. Permite pesquisa de pontos de interesse (POI) por proximidade ou remotas, baseada na orientação da varinha, como um ‘Flashlight’ interactivo e espacial, e a possibilidade dos utilizadores criarem conteúdo adicional para um POI particular como fotos etiquetadas com a localização do POI e a direcção de onde a foto foi tirada. As fotos podem ser filtradas com base num ângulo de visão pretendido num contexto do mundo real.

Premraj et al. [Premraj10] apresentaram o iWalk, uma ferramenta que permite exploração multimédia de dados georeferenciados através: do movimento, para movimentar no espaço digital de uma colecção; e gesto, para a manipulação directa dos dados (exo. *select, go to next, zoom*). Eles fizeram experiências com colecções de fotos e sons georeferenciados, e com uma colecção de museu não georeferenciada, onde o utilizador definia o mapeamento entre os espaços digital e físico. A sua abordagem faz uso de algoritmos de visão por computador executados em tempo-real em câmaras comerciais comuns.

2.4 Interação e Acesso a Vídeo com Segundo Ecrã

Os dispositivos móveis podem também actuar como segundos ecrãs [Courtois12] para complementar e interagir com ecrãs de dimensões superiores como TVs ou até ecrãs em locais públicos. MobiToss, é uma aplicação criada por [Scheible08], que permite a criação e partilha de arte multimédia num grande ecrã público, através de dispositivos móveis com acelerómetros. Os utilizadores podem tirar uma foto ou capturar um vídeo no dispositivo móvel, e “atirá-los” para um grande ecrã público, com um gesto, para visualização e manipulação, através de *tilting*. Os clips criados pelos utilizadores são aumentados ou enriquecidos pelo sistema com items como música ou nomes de marcas e enviados de volta para os seus *smartphones*, como artefactos pessoais do evento.

A avaliação preliminar com utilizadores mostrou que a captura e o atirar do conteúdo móvel para o grande ecrã, bem como a sua manipulação através de gestos na criação da peça de arte foi percebida como uma actividade intuitiva e divertida. Eles gostaram e envolveram-se na experiência, e apreciaram levar algo com eles, especialmente algo com um cariz artístico. Mas a aplicação necessita de melhoramentos, aplicando um conjunto de efeitos de vídeo mais equilibrado, adicionando interacção em grupo e uma interface mais intuitiva, para acomodar

os diferentes gestos que os utilizadores usam para atirar, e aumentar a percepção sobre o que está a acontecer em cada momento. Este trabalho explora gestos naturais com dispositivos móveis para manipular fotos e vídeos no segundo ecrã, mas não explora as dimensões espacial e temporal dos vídeos (e das fotos). E nenhum dos trabalhos relacionados encontrados abordam velocidade e trajectórias em vídeo, como nos propomos fazer.

3. ACESSO A VÍDEO ATRAVÉS DO ESPAÇO, VELOCIDADE E TEMPO

As dimensões espacial e temporal são aqui tidas em conta essencialmente na localização dos vídeos, na forma e velocidade das trajectórias filmadas, e no tempo dentro (exo. ao longo da sua duração) e fora do vídeo (exo. quando foi filmado). Para explorar a interacção e a experiência dos utilizadores com estas dimensões em vídeos georeferenciados, foi desenvolvido um protótipo de alta fidelidade para Android (Fig.1 e 2). Foram concebidas variantes para diferentes cenários de utilização, explorando o uso de sensores e sistemas de localização, na sequência de trabalho preliminar prévio com protótipos de baixa fidelidade [Serra14]. Em seguida, apresentamos o racional subjacente às principais opções de design.



Figura 1. Diferentes modalidades de entrada, através de: a) Gestos (velocidade); b) Toque (velocidade); c) Em “viagem” ou deslocação, exo. num carro; d) Gesto (forma); e) Toque para procura por forma num mapa; e f) e no tempo, por vídeos mais antigos.

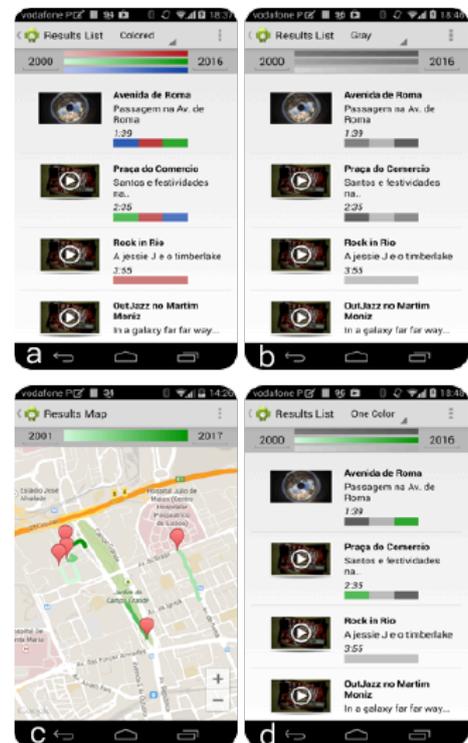


Figura 2. Diferentes modalidades de saída: a) Lista com linha de tempo de 3 cores; b) Lista com linha de tempo de tons de cinzento; c) Resultados num Mapa representando vídeos através das suas trajectórias; e d) Lista com linha de tempo de 1 cor.

3.1 Através de Toque - com o dedo

Esta é a interface mais convencional, que permite desenhar as formas de pesquisa no ecrã através de toque (Fig.1d), ou inclusive num *touch pad* ou rato num portátil ou num PC. A velocidade de realização deste desenho pode também ser capturada para pesquisar apenas por velocidade, ou por ambas: forma e velocidade. Este tipo de interacção pode ser mais familiar e permitir uma maior precisão do que as seguintes, especialmente se o utilizador tiver uma mão livre para fazer esta interacção.

3.2 Através do Gesto - com o telemóvel

Esta modalidade pode se usada movendo o telemóvel (*smartphone*) através de um gesto, para desenhar a forma (Fig.1d) ou demonstrar a velocidade (Fig.1b). A velocidade do gesto é capturada usando o acelerómetro, mas este sensor não permitiu capturar a forma do movimento com precisão suficiente, do modo previsto no protótipo de baixa fidelidade. Assim sendo, foi também usado o giroscópio para capturar a forma do movimento, e o movimento passou a ser efectuado através da inclinação do telemóvel para desenhar a forma como se o cursor fosse uma pequena bola a deslocar-se em cima do ecrã, de acordo com a inclinação feita (Fig.1d).

Esta interacção pode ser efectuada com a mão que segura no telemóvel, deixando a outra mão livre, e tem o potencial para uma modalidade mais natural ou imersiva para seleccionar os vídeos pretendidos, a serem visualizados no telemóvel ou num ecrã de maiores dimensões como uma TV. Neste contexto, os espectadores estão habituados a interagir com um *command* numa só mão, enquanto se mantêm focados no vídeo no grande ecrã.

3.3 Através da Deslocação - em movimento

Quando em deslocação, os utilizadores frequentemente viajam de carro, comboio, metro, avião, etc., ou andam e correm. Em alternativa ou para além do uso da localização actual para aceder a vídeos capturados na mesma localização em que me encontro, pode ser interessante, especialmente quando não se vai a guiar, aproveitar a oportunidade de ver vídeos que foram capturados a uma velocidade análoga à que nos estamos a deslocar, permitindo experienciar o visionamento de forma mais imersiva, fazendo coincidir a velocidade do que estamos a ver com aquela que estamos a sentir fisicamente na realidade (Fig.1c). Isto pode ser particularmente impactante em vídeos de alta velocidade, por exemplo em actividades mais radicais, como corridas de carros, motocross ou sky. Mas também permite por exemplo procurar por vídeos capturados na mesma trajectória que estou a fazer a pé, a uma velocidade análoga, mas há uns anos atrás, o que me permitirá ir comparando a minha experiência actual com o que se podia experienciar nessa altura na mesma localização. Como nos casos anteriores, quer a velocidade quer a forma da trajectória podem ser capturados nesta modalidade. A velocidade e a localização poderão ter um maior potencial de imersão na visualização imediata de vídeos, para obter vídeos análogos ou relacionados com a minha realidade no momento. Mas capturar a trajectória pode também ser interessante para procurar por vídeos que tenha trajectórias semelhantes, mesmo que filmados

em localizações diferentes, por exemplo outras corridas de carros com trajectos semelhantes em qualquer parte do mundo. Enquanto a captura de gestos se baseia em sensores, esta funcionalidade de deslocação, ou viagem, baseia-se em serviços de localização como o GPS.

3.4 Onde - no Mapa or na Localização Actual

Uma vez que os vídeos na nossa aplicação *Sight Surfers* são georeferenciados, as pesquisas podem ser feitas com base na localização. Os utilizadores podem usar um mapa para seleccionar as localizações (Fig.1c), ou podem capturar a localização actual, para as pesquisas (para além de poderem especificar os locais, como por exemplo cidades, através de uma interface mais tradicional baseada em texto).

3.5 Em qualquer Lugar

Os vídeos podem também ser procurados independentemente da sua localização. Para tal, não se faz recurso a mapas, e só a velocidade e a forma são desenhadas no ecrã ou no ar, ou capturadas em deslocação, ou viagem, sem georeferenciação.

3.6 Resultados em Mapas e Listas

Os vídeos resultantes podem ser apresentados como trajectórias num mapa (Fig.2c), onde cada trajectória pode ser vista como uma linha de tempo (*timeline*) do vídeo sincronizada com o vídeo a ser visionado, como no *Sight Surfers* [Noronha12]. É esta a opção por omissão quando a pesquisa é baseada na localização. Mas os resultados também podem ser apresentados de forma independente da sua localização, por exemplo numa lista onde a velocidade (Fig. 2abd) e/ou a forma podem ser evidenciadas na *timeline* de cada vídeo. Note ainda que os utilizadores podem alternar entre as vistas de mapa e lista e seleccionar o que pretendem ver, sobre os mesmos resultados. Os resultados das pesquisas são apresentados com diferentes alternativas de design, evidenciando diferentes aspectos e informações sobre os conteúdos retornados, em termos de forma, velocidade e tempo, quer em mapa ou em lista.

3.6.1 Percepção de velocidade

A velocidade pode variar ao longo de um vídeo, pelo que se devem apresentar primeiro, como resultado, os vídeos que têm a velocidade procurada (com uma tolerância) por mais tempo, e deve dar-se a percepção de quando no vídeo a velocidade é como procurado, e quando é maior e menor. A Fig.2abd apresenta três alternativas de design para a apresentação da velocidade nas linhas de tempo dos vídeos, através de: a) 3 Cores: verde para a velocidade procurada, vermelho para mais rápido e azul para mais lento (RGB); b) Tons de Cinzento: um tom intermédia para a velocidade procurada, mais escuro para mais rápido e mais claro para mais lento; and c) 1 Cor: verde para a velocidade procurada e dois tons de cinzento para mais rápido e mais lento, como em b), permitindo um maior contraste e destaque para a cor procurada.

3.6.2 Percepção de forma

A forma é apresentada por omissão na vista de mapa, mas também pode ser apresentada na lista, como na Fig.2c, onde as linhas de tempo de cada vídeo tomam a

forma da trajectória correspondente. A percepção de velocidade é opcional em todas as linhas de tempo: mapa e lista, com ou sem forma.

3.6.3 Percepção e procura por tempo

Para além de se apresentar a duração junto de cada vídeo, e o tempo corrente na linha de tempo, o tempo em que o vídeo foi capturado é representado pela cor, quer da linha de tempo nas listas (Fig.2abd) quer das trajectórias nos mapas (Fig.2c), para representar a sua idade. Optamos por perder saturação para os vídeos mais velhos, e manter a vivacidade da cor para os mais novos, de forma proporcional.

No topo do ecrã, uma linha de tempo apresenta a escala de tempo de referência, colorida de forma correspondente à “idade” dos vídeos. A menor saturação à esquerda, para os mais antigos, que vai gradualmente aumentando para a direita, em direcção aos mais recentes. As etiquetas de tempo (um ano no exemplo da Fig.2) que representam o intervalo de tempo são apresentadas em caixas de texto à esquerda e direita da linha de tempo. Esta linha torna explícito o mapeamento entre as cor e o tempo, permitindo a identificação dos vídeos mais recentes e mais antigos no intervalo de tempo em causa. Esta linha de tempo pode ter apenas uma cor, quando a velocidade não é apresentada (por exemplo, apenas verde no mapa da Fig.2c), ou ter três barras, uma por cada categoria de velocidade (por exemplo, tons de cinzento ou RGB nas listas das Figs.2ba) para evidenciar como as diferentes cores “envelhecem”.

Nesta vista, os utilizadores podem procurar ou filtrar vídeos por tempo: quer seleccionando por toque as duas caixas de texto das etiquetas, e escrevendo o ano pretendido num teclado virtual; quer usando gestos para incrementar ou decrementar o seu valor, desenhando uma espiral com o dedo no ecrã (o gesto evidenciado no desenho na Fig1.d). Uma espiral no sentido dos ponteiros dos relógios significa avançar no tempo, enquanto que no sentido contrário significa recuar. Este mapeamento foi escolhido com base numa hipótese de que seria o mapeamento mais natural, considerando a metáfora dos relógios de ponteiros.

4. AVALIAÇÃO COM UTILIZADORES

Foi realizada uma avaliação com utilizadores para avaliar as funcionalidades concebidas em termos da sua utilidade, experiência de utilização e usabilidade, para investigar sobre as alternativas mais eficazes e preferidas, e sobre a sua aplicação em cenários de uso reais.

4.1 Método

Foi realizada uma avaliação baseada em tarefas, executadas pelos utilizadores participantes, recorrendo a Observação e Entrevistas semi-estruturadas, depois de se explicar o objectivo da avaliação, o conceito subjacente ao contexto da aplicação Sight Surfers, e as funcionalidades em avaliação no protótipo de alta fidelidade. A ordem das variantes foi alterada para cada utilizador, para que esta não viesasse os resultados, seguindo uma amostragem por medidas repetidas, onde todos os utilizadores testaram todas as variantes do protótipo.

No final de cada tarefa, os utilizadores atribuíram uma classificação baseada no questionário USE (Usefulness, Satisfaction, and Ease of use [Lund01], que captura as dimensões essenciais de utilidade, experiência de utilização e usabilidade) utilizando uma escala de 1-5, para cada funcionalidade interactiva testada, em todas as suas variantes, e encorajados a tecer comentários e apresentar sugestões. As avaliações tiveram lugar no campus da nossa universidade, maioritariamente em espaços interiores, onde os utilizadores, acompanhados pelo avaliador, podiam sentar-se ou caminhar.

4.2 Participantes

A avaliação contou com 9 utilizadores participantes com idades compreendidas entre 18 e 27 anos (22,5 em média, 3 do sexo feminino e 6 do sexo masculino). Todos os utilizadores haviam no mínimo terminado a escola secundária, 3 estavam na área de informática, os restantes de áreas diversas, todos tinham um *smartphone* usado diariamente para aceder a informação, e 9 faziam diariamente pesquisas e visionamentos de vídeos mas maioritariamente em PCs, por vezes em *tablets*, e mais raramente em telemóvel.

4.3 Resultados

Os principais resultados são sumarizados através dos valores médios do USE e dos comentários mais significativos, em cada categoria de funcionalidades.

4.3.1 Procura por velocidade

Foi pedido aos utilizadores participantes que “procurassem vídeos por velocidade, usando toque e gestos”. Através de toque: foi considerado bastante divertido, muito fácil de usar, e útil pela maioria dos utilizadores (U: 3.67; S: 3.78; E: 4.44). Exemplo de comentário: “*A pesquisa georeferenciada é sem dúvida útil em muitas situações, por exemplo para conhecer certas zona de uma cidade*”, “*Eu imagino um atleta a usar esta funcionalidade para procurar pistas de corrida*”. Através de gestos: teve classificações ligeiramente mais baixas, especialmente na facilidade de uso (U: 3.22; S: 3.22; E: 3.67), nesta nova modalidade, menos familiar.

Finalmente, foi pedido aos utilizadores que “procurassem vídeo usando a velocidade de ‘viagem’ ou deslocação”. Foi-lhes pedido que “procurassem por vídeos com uma velocidade correspondente a um passo rápido”, que eles realizaram andando. Os resultados foram satisfatórios (U: 3.43; S: 3.0; E: 4.11), mas alguns utilizadores mencionaram que esta funcionalidade poderia ser mais útil quando em viagem por exemplo de carro ou comboio, e que se tinha sentido pouco à-vontade a caminhar rapidamente para a avaliação, não se sentindo suficientemente confortáveis e relaxados para depois ver os vídeos, enquanto se mantinham a caminhar com a mesma velocidade. Outros ambientes e contextos de deslocação e viagem serão testados em avaliações futuras.

4.3.2 Procura por forma

Foi solicitado aos utilizadores que “procurassem por uma dada forma numa localização específica no mapa através de toque (usando o dedo)”. Esta funcionalidade foi bem recebida (U: 3.44; S:3.11; E:3.44): os participantes apre-

ciaram ter a capacidade de georreferenciar os trajectos representados pelas formas, e encontraram vários cenários onde poderia ser usada: por exemplo: “*Como atleta, eu poderia usar esta pesquisa para explorar melhor pistas de corrida com certos formatos, e vê-las no mapa*”, “*A procura georreferenciada é sem dúvida útil em muitas situações, por exemplo para conhecer melhor certas partes da cidade*”.

A tarefa “Procure por vídeo com o formato em ‘u’, usando o telemóvel para desenhar a forma com um gesto” foi considerada muito difícil e pouco útil, tendo sido a que teve pior aceitação por parte dos participantes (U: 1.89; S: 2.22; E: 2.22), “*Eu não considero que seja prático de realizar esta procura enquanto estou a fazer outras actividades, como a andar ou até a falar com outra pessoa*”. O equilíbrio entre a velocidade e a sensação de controlo é na nossa opinião longe de ser óptima, no estado actual do protótipo, não nos permitindo criar o tipo de experiência que tínhamos em mente para esta funcionalidade. Para além disso, os utilizadores não estão habituados a usar o telemóvel para este tipo de acção ou interacção, tendendo a preferir usar o dedo para desenhar.

4.3.3 Resultados em Mapas e Listas

Em geral, os utilizadores preferem a forma mais familiar de apresentar vídeos numa **lista** (U: 4.0; S: 3.56; E: 4.44), por ser fácil e simples, “*É a forma mais rápida de ver vídeos, se a localização geográfica não for relevante*”. De qualquer forma, o **mapa** também teve boa aceitação (U: 3.44; S: 3.11; E: 3.78), e a maioria dos participantes referiram que estes eram úteis quando se queria procurar por trajectórias georeferenciadas, ou por vídeos em locais específicos, ou quando nos queremos aperceber das localizações e dimensões das trajectórias subjacentes aos vídeos. Também comentaram que: “*É interessante ver as trajectórias dos vídeos [no mapa] sem a necessidade de visionary o vídeo*”.

As principais preocupações mencionadas dizem respeito à percepção da quantidade de vídeos recuperados e a sua representação no caso de se uma quantidade muito elevada. A filtragem dos resultados não estava no âmbito deste teste, mas está alinhada com as nossas próprias questões de investigação mais abrangentes, e já abordadas inclusivamente em trabalhos anteriores [Ramalho13a,13b].

Houve ainda sugestões no sentido de ter uma mistura das vistas de lista e mapa, onde ao navegar uma lista de vídeos, poderíamos ver a sua localização num mapa; e quando passássemos o cursor por cima de uma trajectória no mapa, poderíamos ver informação mais detalhada sobre os vídeos nessa trajectória numa lista (com velocidade, duração, uma imagem representativa do vídeo, título, data etc.).

Em termos de **percepção e procura por tempo**, as linhas de tempo (*timelines*) foram consideradas úteis, satisfatórias, e fáceis de usar para encontrar a velocidade procurada nos vídeos resultantes. Apresentamos três designs alternativos, e a variante preferida foi a de “1 cor” (tabela 1). A maioria dos utilizadores preferiram este modo (U: 4.11; S: 4.11; E: 4.44) em que se evidencia a velocidade

procurada numa cor (nesta caso, em verde), enquanto as outras velocidades são menos evidenciadas (em cinzento), por ser mais fácil de usar. Em seguida preferiram a variante de “3 cores” (U: 3.44; S: 3.44; E: 3.6), que consideraram útil e bastante satisfatória, mas mais difícil de usar: “*A linha de tempo com uma cor é uma forma fácil e directa de encontrar a velocidade procurada. A linha de tempo com 3 cores é mais difícil de perceber e torna a interface menos clara, mais cheia e berrante*”. A variante em tons de cinzento foi considerada mais difícil de usar e menos útil, satisfatória e até menos divertida (U: 2.56; S: 2.44; E: 2.67). Estas preferências em relação aos designs alternativos para a percepção de velocidade condizem com as nossas expectativas, que correspondiam às nossas hipóteses descritas na fase de design.

designs	Utilidade		Satisfação		Facil. d/ Uso	
	M	σ	M	σ	M	σ
1 Cor	4,11	0,31	4,11	0,5	4,44	0,83
Cinzas	2,56	0,68	2,44	0,5	2,67	0,67
3 Cores	3,44	0,76	3,44	0,68	3,44	0,88

Table 1: Percepção de velocidade em trajectórias nas linhas de tempo.

Depois da procura, foi solicitado aos utilizadores que filtrassem os resultados “recuando no tempo para vídeo mais antigos”, em duas modalidades de interacção: usando o teclado virtual para escrever o ano, e gestos através de toque. Os resultados indicam que os utilizadores gostaram de ambas as modalidades para filtrar os resultados pelo tempo, no entanto, consideraram o teclado ligeiramente mais útil e fácil de usar (U: 3.67; S: 3.22; E: 4.0), enquanto que os gestos foram considerados mais satisfatórios e divertidos (U: 3.33; S: 3.67; E: 3.44). Numa perspectiva mais geral, os utilizadores consideraram que as duas modalidades se complementam, sendo os gestos mais rápidos e o teclado mais preciso.

5. CONCLUSÕES E TRABALHO FUTURO

Neste artigo apresentou-se a motivação e as opções de design para o acesso multimodal e móvel a vídeo georeferenciado, no espaço e no tempo, num contexto maioritariamente de conteúdo gerado pelos utilizadores. Estas ideias foram materializadas num protótipo de alta fidelidade que foi desenvolvido para Android e que foi avaliado com utilizadores, em termos da sua utilidade, facilidade de uso, satisfação e preferências.

O nosso foco assentou em três dimensões principais: forma, velocidade e tempo, e explorou a multimodalidade potenciada pelos telemóveis mais recentes (*smartphones*) para conceber novas formas de aceder e navegar em vídeos georeferenciados, com base em toque, gestos e movimento.

A avaliação com utilizadores foi encorajadora. Os utilizadores consideraram a maioria das funcionalidades bastante satisfatórias, até divertidas, e fáceis de usar, e as diferentes opções de design e modalidades de interacção foram consideradas interessantes e adequadas para

diferentes cenários de utilização que foram identificados. A utilidade e a facilidade de uso foram mais rapidamente associadas às modalidades mais familiares, enquanto que os gestos e o movimento, ou deslocação, potencialmente mais naturais foram considerados bastante satisfatórias e por vezes até divertidas. Em particular, os gestos para navegar no tempo, através de vídeos mais antigos e mais recentes, foram muito bem recebidos pelos participantes, e foram apontados como uma forma eficaz e rápida de procurar e navegar no tempo, que podia combinar bem com uma modalidade mais precisa, versátil e tradicional de introduzir um tempo específico, usando um teclado.

Por outro lado, a funcionalidade menos apreciada foi a procura por forma através de gestos com o telemóvel, que por razões técnicas assentou numa implementação que não permitia tanto controlo e precisão como havia sido concebido em baixa fidelidade. Adoptou a inclinação do telemóvel, em vez de permitir executar um gesto no ar com o telemóvel na mão, desenhando no ar a forma pretendida, o que também permitiria uma melhor captura da velocidade do movimento, para a procura também por velocidade. Esta foi considerada uma limitação do actual protótipo logo à partida, e ela veio a reflectir-se nos resultados, de acordo com a nossa expectativa.

Outro aspecto que pode vir a ser melhorado refere-se às condições de avaliação em cenários mais realistas, em que os utilizadores possam sentir mais de perto a necessidade ou os benefícios de usar trajectórias e a velocidade, para aceder aos vídeos nas diferentes modalidades e na “vida real”, como as que alguns identificaram nos comentários, mas que a maioria não está habituada a vivenciar.

Os próximos passos incluem o refinar e a extensão de algumas das soluções actuais, baseados no retorno obtido dos utilizadores, e explorar novas formas de navegar, por exemplo, no tempo, e também no próprio visionamento imersivo do vídeo. A área dos segundos ecrãs é também uma área que estamos a abordar, e algum trabalho está já a ser desenvolvido para se poder, por exemplo, “atirar” o vídeo para um ecrã de maiores dimensões, como uma TV. No grande ecrã pode-se visionar o vídeo de uma forma mais imersiva, enquanto se pode continuar a usar o dispositivo móvel para ajudar a orientação, a percepção e a navegação, inclusivé com a sincronização com um mapa e as opções de selecção, e para aceder a conteúdo mais detalhado ou relacionado. Estas direcções de investigação alinham com os desenvolvimentos mais recentes e as tendências na área de TV interactiva e vídeo online, crescentemente acedido através de um ecossistema com vários dispositivos, incluindo os móveis [Obrist15].

6. AGRADECIMENTOS

Este trabalho foi parcialmente suportado pela FCT através do Projeto Estratégico LaSIGE, com referência PEst-OE/EEI/UI0408/2014; o Projecto de investigação ImTV com referência UTA-Est/MAI/0010/2009; e a bolsa de doutoramento FCT dentro do programa UT Austin|Portugal com referência SFRH/BD/51798/2011.

7. REFERÊNCIAS

- [Courtois12] Courtois, C., D’heer, E. Second screen applications and tablet users: constellation, awareness, experience, and interest. *Proc. EuroITV’12*, 2012, ACM Press, 153-156.
- [Finsterwald12] Finsterwald, J. M., Grefenstette, G., Law-To, J., Bouchard, H., and Mezaour, A.D. The Movie Mashup Application MoMa: Geolocating and Finding Movies. *Proc. GeoMM’12*, 2012 ACM, 15-18.
- [Hao11] Hao, J., Wang, G., Seo, B., and Zimmermann, R. Keyframe Presentation for Browsing of User-generated Videos on Map Interfaces. *Proc. ACM MM’11*, 2011, ACM Press, 1013-1016.
- [Lei09] Lei, Z., and Coulton, P. A. Mobile Geo-wand Enabling Gesture Based POI Search an User Generated Directional POI Photography. *Proc. ACE’09, Int. Conf. on Advances in Computer Entertainment Technology*, 2009, ACM, 392-395.
- [Lund01] Lund, A.M. Measuring usability with the USE questionnaire. *Usability and User Experience*, 8(2), 2001.
- [Norman93] Norman, D. *Things that Make us Smart*. Addison Wesley Publishing Company, 1993.
- [Noronha12] G. Noronha, C. Álvares, T. Chambel. Sight Surfers: 360° Videos and Maps Navigation. *Proc. Geo MM’12*, 2012, ACM Press, 19-22.
- [Obrist15] Obrist, M., Cesar, P., Geerts, D., Bartindale, T., and Churchill, E. Online Video and Interactive TV Experiences. *Interactions*, ACM, Sep-Oct 2015, pp. 33-37.
- [Premraj10] Premraj, V., Schedel, M., and Berg, T.L. iWalk, A Tool for Interacting with Geo-Located Data Through Movement and Gesture. *Proc. ACM MM’10*, 2010, ACM Press, 1059-1062.
- [Ramalho13a] Ramalho, J. and Chambel, T. Windy Sight Surfers: sensing and awareness of 360° immersive videos on the move. In *Proc. of EuroITV 2013*, Como, Italy. 107-116.
- [Ramalho13b] J. Ramalho, Chambel, T. Immersive 360° Mobile Video with an Emotional Perspective. In *Proc. ImmersiveMe’2013* (ACM Press 2013) 35-40.
- [Rego07] Rego, A., Baptista, C., Silva, E., Schiel, U., and Figueirêdo, H. VideoLib: a Video Digital Library with Support to Spatial and Temporal Dimensions. In *Proc. SAC’07*, 2007, ACM, 1074-1078.
- [Scheible 08] Scheible, J., Ojala, T., and Coulton, P. Mo-biToss: A novel gesture based interface for creating and sharing mobile multimedia art on large public displays. In *Proc. ACM MM’08*, 2008, ACM Press, 957-960.
- [Serra14] Serra, S., and Chambel, T. Towards Multimodal Consumption of Georeferenced Mobile Video Using Shape and Speed. In *Proc. of 2nd Int. Workshop on Interactive Content Consumption* at ACM TVX 2014.

Visualização tri-dimensional interativa de informação demográfica mundial

Eduardo Duarte Pedro Bordonhos
DETI, Universidade de Aveiro
{emod,bordonhos}@ua.pt

Paulo Dias Beatriz Sousa Santos
DETI, IEETA, Universidade de Aveiro
{paulo.dias,bss}@ua.pt

Resumo

Este artigo propõe o Living Globe, uma aplicação de visualização de informação tri-dimensional que apresenta dados demográficos mundiais. A aplicação permite a visualização simultânea de vários dados demográficos, tais como a população, a densidade e o crescimento populacional, a esperança de vida, o saldo migratório, a taxa bruta de natalidade/mortalidade e o rácio entre estas duas taxas. O mapeamento dos dados para as estruturas visuais na aplicação é configurável pelo utilizador, introduzindo um aspeto de personalização que incentiva a obtenção somente visual e mais imediata de informação. Adicionalmente, os dados representados são normalizados conforme uma escala configurável, permitindo filtrar amostras muito dispare (como por exemplo na China e na Índia) e aumentando assim a visibilidade dos restantes dados. Foram realizados testes preliminares que indicam que a filtragem dos dados e a personalização no mapeamento incentivam uma leitura mais rápida de informação relacional e que a pesquisa e seleção de países tornam a navegação mais intuitiva comparado com as visualizações estado-de-arte.

Palavras-Chave

living globe, visualização de informação, tri-dimensional, demografia, população, filtragem, indicadores

1. INTRODUÇÃO

A visualização de dados demográficos é um método popular em estudos científicos para inferir e apresentar o crescimento populacional. Muitos destes estudos apresentam esta informação recolhida ao longo de vários anos em mapas bi-dimensionais e tri-dimensionais, para demonstrar os diferentes níveis de crescimento que ocorrem em várias localizações no globo terrestre.

Este artigo propõe o Living Globe, uma aplicação de visualização com o objetivo de incentivar a investigação e observação de dados demográficos de forma intuitiva e permitir aos utilizadores a comparação de dados de vários países em simultâneo. Esta aplicação implementa funcionalidades não-existentes nas visualizações estado-de-arte e que facilitam a navegação pelo globo, tais como o controlo e ampliação da câmara tri-dimensional, a seleção e a procura textual de países através de um sistema de pesquisa com preenchimento automático.

Nesta visualização estão representadas a população, a densidade e o crescimento populacional, a esperança de vida, o saldo migratório, a taxa bruta de natalidade e a taxa bruta de mortalidade. É possível filtrar esta informação conforme um valor máximo e um valor mínimo, podendo por isso retirar países com valores extremamente grandes ou pequenos, quando comparado com a moda estatística, e reconfigurar a visualização para melhorar a comparação

de dados entre diferentes países.

De modo a possibilitar a representação gráfica de dados diferentes em simultâneo, para inferir possíveis relações entre diferentes indicadores, o protótipo desenvolvido permite o mapeamento de dados sobre três componentes: altura de paralelepípedos/barras verticais em posições geográficas, cor destas barras e cor dos países. A natureza tridimensional da visualização permite o mapeamento de dois tipos de dados sobre as barras verticais, quanto à sua forma e quanto à sua cor. As cores das barras e dos países representam uma gama de valores normalizados entre 0 e 1, sendo a cor obtida através da seleção numa escala de cores.

Este artigo está organizado da seguinte forma: a secção 2 descreve as visualizações existentes no domínio da representação tri-dimensional de dados demográficos; a secção 3 descreve a visão por trás da aplicação proposta e aponta as funcionalidades ausentes de acordo com essa visão; a secção 4 aborda a implementação técnica da visualização, incluindo a fase inicial de recolha e tratamento de dados; a secção 5 descreve os testes realizados à aplicação para determinar a sua mais-valia quanto às visualizações estado-de-arte; e por fim, a secção 6 apresenta conclusões obtidas a partir dos testes da secção 5 e indica aspetos a desenvolver para trabalho futuro.

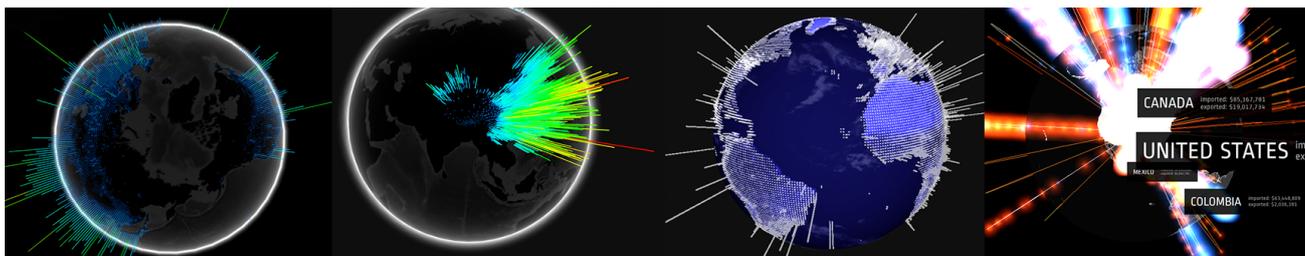


Figura 1. Visualizações estado de arte. Da esquerda para a direita: 'WebGL Globe - World Population', 'WebGL Globe - China and US Population', 'World Population Density - 2010' e 'Small Arms and Ammunition'.

2. TRABALHO RELACIONADO

Nesta secção são apresentadas outras soluções de visualização de informação demográfica. Todas as visualizações do domínio, distribuídas de forma gratuita e *open-source*, são implementadas como aplicações web e utilizam a tecnologia WebGL ou outras APIs de representação tri-dimensional em Javascript, como *three.js*¹. O uso destas tecnologias permite: i) a criação e processamento gráfico de objetos tri-dimensionais, juntamente com as suas texturas, *canvas* e *shaders*, e manipuláveis em tempo-real; e ii) torna a solução compatível com um número elevado de dispositivos, necessitando apenas de conexão de rede e de um *browser* que suporte estas tecnologias, tal como os *browsers* mais populares no mercado. Todas as visualizações demográficas estudadas distribuem um tipo de dados, população total ou densidade populacional, por localização latitude-longitude ao longo de um mapa mundial, sob a forma de barras verticais posicionadas nessas localizações. Quanto maior a população ou densidade naquela localização, maior será essa barra.

Uma destas visualizações é o WebGL Globe², um globo implementado com a API nativa do WebGL e realizado no âmbito do programa "Chrome Experiments". O WebGL Globe é uma plataforma onde qualquer desenvolvedor pode fornecer um conjunto de dados com posições geográficas conforme uma API estabelecida neste e obter uma visualização num globo tri-dimensional, sendo assim totalmente abstraído da implementação deste globo e de elementos gráficos adicionais. Dos projetos WebGL Globe disponibilizados no website oficial, selecionamos dois projetos que apresentam dados demográficos: "World Population"[Arts 11] e "China and US Population"[Zhifang 13].

"World Population", que utiliza dados disponibilizados pelo Centro de Dados e Aplicações Socio-económicas (SE-DAC), é muito limitado em termos de usabilidade e estudo de informação, visto que a única funcionalidade disponível é o movimento do globo. A única componente representativa de dados é a altura das barras verticais, pelo que esta visualização não tira partido das suas três dimensões.

"China and US Population" utiliza um conjunto de dados mais limitado que o anterior, apresentando dados mais específicos para a China e os Estados Unidos da América. No que toca à interação e à usabilidade, estas são idênticas ao trabalho anterior, sofrendo por isso das mesmas falhas e tendo ausente o aproveitamento da oportunidade tri-dimensional.

Uma alternativa ao WebGL Globe é o "World Population Density - 2010"[Dirksen 12], um globo implementado em *three.js*. Identicamente ao "World Population", os dados utilizados são recolhidos do SEDA, mas em contraste com este, globo não é manipulável, rodando em torno do eixo de *yy* automaticamente. Deste modo, a usabilidade é menor pois um utilizador é forçado a esperar que o globo rode para observar os dados de um dado país e tem um tempo limitado para realizar esta observação. Mais uma vez, esta visualização não introduz benefícios na utilização de três dimensões.

Apesar de não fazer parte do domínio de visualização de dados demográficos, foi estudada a visualização "Small Arms and Ammunition - Imports & Exports"[Arts 14]. Os dados desta correspondem ao comércio de armas e munições de pequeno-porte entre os anos 1992 e 2010. Em contraste com as visualizações anteriores, este implementa a seleção e a pesquisa com preenchimento automático de países, para possibilitar a descoberta de países com localização desconhecida pelo utilizador. O tráfico entre dois países é representado com uma linha que interliga esses países. Estas linhas de ligação tiram partido da terceira dimensão para orbitar à volta do globo, não ocultando graficamente os países e o *feedback* de seleção. Deste modo, a tri-dimensionalidade da visualização introduz uma contribuição não apenas estética.

3. CONCEÇÃO DA VISUALIZAÇÃO

Nesta secção descrevemos o modelo da visualização proposta, referindo as funcionalidades que estão em falta de acordo com este modelo e os principais contributos que a aplicação vem trazer para o domínio em que se insere.

3.1. Motivação

As visualizações estudadas dentro do domínio de visualização de informação demográfica apresentam limitações ao nível da interação quando comparadas com

¹<http://threejs.org>

²<https://www.chromeexperiments.com/globe>

uma visualização fora deste domínio. O "Small Arms and Ammunition - Imports & Exports" implementa funcionalidades de seleção e pesquisa de países, funcionalidades que poderiam ser facilmente transpostas para as restantes visualizações, melhorando a usabilidade destas.

Adicionalmente, não são introduzidas mais-valias quanto ao uso de três dimensões, visto que os seus componentes tri-dimensionais poderiam ser substituídos por bi-dimensionais sem perda de legibilidade (por exemplo, substituir paralelepípedos, onde valores são representados pela sua altura, por pie-charts, onde valores são representados pelo seu preenchimento).

Todas as visualizações tri-dimensionais no domínio da demografia adotam o modelo de referência de visualização de informação [Chi 98][Mazza 09] para implicitamente transformar e mapear dados, dando ao utilizador a possibilidade de interação e configuração somente sobre a apresentação, ou seja, a última fase do modelo de referência. O mapeamento dos dados em estruturas visuais é feito implicitamente e sem possibilidade de configuração, o que implica que a leitura dos dados está cingida às opções de design estabelecidas pelos desenvolvedores. Numa perspetiva onde um utilizador utiliza estas ferramentas para estudo demográfico, onde este poderá procurar por relações entre diferentes tipos dados para inferir se o efeito de um indicador é a causa de perturbações noutra, seria interessante permitir uma maior flexibilidade na fase de mapeamento para apropriar a leitura e descoberta destas relações às preferências do utilizador. Juntamente com este aspeto, porque o utilizador teria a possibilidade de selecionar o tipo de dados que pretende consultar, seria benéfico poder observar outros tipos de dados para além da população total ou a densidade.

Por fim, todas estas visualizações apresentam a informação de forma estática, não permitindo realizar um *zoom* sobre a informação para melhorar a legibilidade dos dados relevantes. Em consequência, grande parte dos dados populacionais tornam-se pouco legíveis devido à existência de um pequeno conjunto de casos com valores extremamente elevados ou extremamente baixos quando comparados com os restantes dados. Isto verifica-se, por exemplo, devido ao facto da China e a Índia possuírem juntamente aproximadamente 36% da população mundial, ou do Mónaco possuir uma grande densidade populacional quando comparado com outros países, implicando que as alturas das barras verticais de países com população média e baixa sejam extremamente pequenas.

3.2. Contributos e Modelo

Como ponto de partida, propõe-se a implementação de funcionalidades de usabilidade existentes na visualização "Small Arms and Ammunition - Imports & Exports". Deste modo, a seleção permite a apresentação detalhada dos dados do país selecionado de modo textual e a pesquisa por nome de países permite a seleção de um país cuja localização é desconhecida pelo utilizador.

O Living Globe deverá naturalmente corresponder aos mo-

delos de visualização referidos, ao receber dados, transformar, mapear e apresentar estes. No entanto, apesar de fazer algum mapeamento por predefinição, o seu principal contributo está no facto de fornecer ao utilizador a possibilidade de realizar o seu próprio *mapping*. Para isto, conceberam-se três estruturas visuais, a que chamamos de Componentes, e cuja representação é configurável pelo utilizador: i) a altura de barras verticais, onde a menor altura é o valor mais baixo e a maior altura é o valor mais alto; ii) a cor das barras verticais, que varia entre um conjunto de cores de uma escala de azul a amarelo, onde azul é o valor mais baixo e amarelo é o valor mais alto; e iii) a cor dos países, que varia entre um conjunto de cores de uma escala de vermelho a verde, onde vermelho é o valor mais baixo e verde é o valor mais alto. O uso de três dimensões permite que um mesmo elemento gráfico, neste caso, a barra vertical, possa ser utilizado quanto à sua forma e a sua cor para representar dois tipos de dados.

A escolha dos pares cores azul-amarelo e vermelho-verde parte do princípio da harmonia de cores, visto que azul é cor complementar de amarelo e vermelho é cor complementar de verde (e vice-versa). As cores complementares, em termos de semiótica e percepção de cor, são ideais para representar ideias opostas, como por exemplo, direções cardinais, valores menores e maiores, entre outros [Pridmore 11]. Optou-se por dois pares de cores distintos em vez de um par para melhor representar Componentes (e portanto Indicadores) distintos.

Através desta funcionalidade, o utilizador deverá escolher qual o tipo de informação que cada Componente irá representar, entre um conjunto de tipos de informação possibilitados pelos dados originais recolhidos. Estes tipos de dados que denominamos Indicadores correspondem na aplicação desenvolvida a: i) população total; ii) densidade populacional; iii) crescimento populacional; iv) esperança média de vida; v) saldo migratório; vi) taxa bruta de natalidade (*Crude Birth Rate*, ou CBR); vii) taxa bruta de mortalidade (*Crude Death Rate*, ou CDR); viii) diferença entre o CBR e o CDR. O número de Indicadores que podem ser representados em simultâneo neste modelo é maior que o número de Indicadores disponíveis nas visualizações estado-de-arte mencionadas, viabilizando a procura de relações causa-efeito entre Indicadores diferentes.

Como referido anteriormente, um problema das visualizações analisadas é a dificuldade de visualização em situações em que existem valores máximos extremamente maiores (e/ou valores mínimos extremamente menores) do que a moda estatística da amostra. Para aliviar este problema, o Living Globe implementa uma funcionalidade de Filtragem onde utilizador estabelece o limite máximo e mínimo da amostra, definindo uma nova gama de valores de um Indicador e excluindo países com valores maiores ou menores que estes limites. Por exemplo, ao filtrar um país reduzindo o valor máximo, os valores dos restantes países são normalizados à nova escala de valores sem este país, aumentando por isso a altura das barras ou



Figura 2. "Living Globe" a apresentar dados demográficos de 2010 e com Reino Unido selecionado, mostrando detalhes textuais deste no canto inferior esquerdo.

aproximando as cores do amarelo, no caso das barras, e do verde, no caso dos países. Isto permite que representações de componentes semelhantes possam potencialmente ter uma maior diferenciação, sendo por isso mais facilmente comparáveis e legíveis visualmente.

A possibilidade de observar dados de diferentes anos para poder acompanhar o seu crescimento ou decréscimo ao longo do tempo é um aspeto importante na comunidade científica para a descoberta de fenómenos e a inferência de comportamento futuro. Deste modo, o modelo aqui proposto assume a existência de valores para cada Indicador e para ARs (anos de recolha) diferentes, e por isso o protótipo que o suporta apresenta uma estrutura visual que permite ao utilizador definir o AR em visualização.

4. IMPLEMENTAÇÃO DO PROTÓTIPO

Nesta secção iremos descrever um protótipo que implementa o modelo descrito na secção anterior, utilizando a linguagem Javascript, a biblioteca *three.js*, e outras bibliotecas como *chroma.js*³ e *jQuery*⁴. Este protótipo foi testado com sucesso nos browsers Firefox 37.0.1, Chrome 42.0.2311.90 (64-bit) e Safari 8.0.5, e está disponível livremente em <https://github.com/eduardo/living-globe>.

4.1. Recolha e leitura de dados

Para conceber a visualização proposta, seria necessário recolher um conjunto de dados com os diversos Indicadores por país e por AR. Foram obtidos dados a partir do World Bank⁵, com *open license* para uso não-comercial. Estes dados estão estruturados como uma tabela, tendo por isso uma linha por cada par País-Indicador e uma coluna por cada AR. Os ARs destes dados vão desde 1980 até 2014, sendo que o Indicador de saldo migratório é apenas registado a cada 5 anos. Por fim, os países são identificados pelo nome oficial e pelo código ISO 3166 alpha-3⁶, um

³<http://old.driven-by-data.net/about/chromajs/>

⁴<https://jquery.com>

⁵<http://data.worldbank.org>

⁶http://www.iso.org/iso/country_codes

código que identifica unicamente cada país com três letras.

Tendo já os dados dos Indicadores para um período de tempo, resta obter a localização espacial dos países, representados por pares de valores latitude-longitude, para posicionar a barra vertical. Esta informação foi obtida a partir do *Sokrata Open Data Portal*⁷, um portal de dados open-source, também para uso não-comercial. Os dados obtidos correspondem a uma tabela com colunas para o código ISO 3166 alpha-3 e para a latitude e longitude, entre outros.

Todos os dados recolhidos são lidos assincronamente utilizando a tecnologia Ajax e analisado de modo a descobrir os valores máximos e mínimos dos Indicadores selecionados. Estes valores são utilizados para normalizar os restantes dados, como é descrito na sub-secção "Globo e Países" e "Barras Verticais".

4.2. Globo e Países

Outras funcionalidades propostas nesta visualização são a seleção e coloração de países. A implementação destas funcionalidades implica um conjunto de requisitos como o desenho e identificação dos países num objeto tri-dimensional, para possibilitar a deteção de seleção e a modificação da cor de cada país individual.

Como primeira solução para identificar os países no globo, utilizou-se um ficheiro CSV (*Comma Separated Values*)⁸ que contém um conjunto de pares latitude-longitude para cada país, sendo que o conjunto de pontos geográficos representa um polígono equivalente ao território de um dado país. O uso deste ficheiro implica o desenho programático dos países de modo a poder delimitá-los no globo, permitindo colorir os vários polígonos tal como requerido. Para identificar um país selecionado, a localização do cursor do rato era traduzida para latitude-longitude, sendo comparada com os limites do país. No entanto esta implementação mostrou-se como muito pouco eficiente pois o desenho dos múltiplos polígonos que se alteram quando se reconfiguram os Indicadores, o AR e/ou a Filtragem tem um custo computacional excessivo e torna a visualização demasiado lenta, afetando consideravelmente a experiência de utilização.

A segunda solução considerada consiste no uso de texturas, *canvas* e um *raycaster*, inspirado na implementação da visualização "Small Arms and Ammunition"⁹. São sobrepostas no globo cinco texturas diferentes: a) uma textura *satellite*, com uma fotografia da terra, utilizada apenas para efeitos estéticos; b) uma textura *specular*, que ajuda a manter a luminosidade dos países em relação ao mar; c) uma textura *ratio* inicialmente transparente, onde dadas áreas são coloridas com a cor correspondente ao valor normalizado do Indicador escolhido; d) uma textura *select* inicialmente transparente, onde dadas áreas são coloridas de branco, um de cada vez, para servir de feedback do país selecionado; e) uma textura *lookup*, que contém uma imagem

⁷<https://opendata.sokrata.com/dataset/Country-List-ISO-3166-Codes-Latitude-Longitude/mnkm-8ram>

⁸<https://mledoze.github.io/countries/>

⁹<http://mflux.tumblr.com/post/28367579774/armstradeviz>

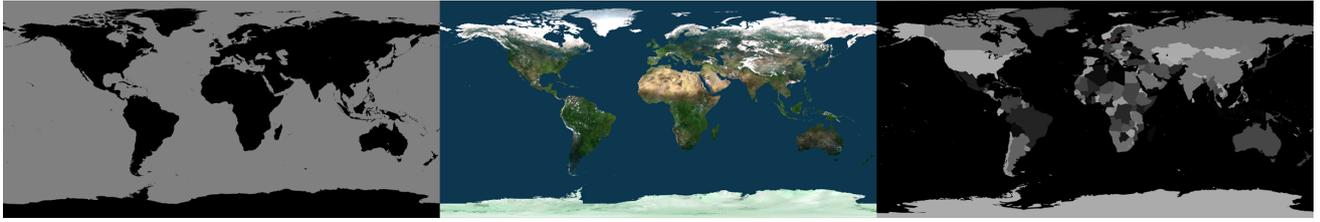


Figura 3. As três texturas utilizadas no material do globo. Da esquerda para a direita: textura *specular*, textura *satellite* e textura *lookup*

com o plano terrestre numa escala de cinzentos, onde cada país está colorido com um grau de luminosidade de cinzento diferente. Deste modo, esta textura *lookup* introduz as áreas que cada país cobre, e possibilita a sobreposição de cores sobre essas áreas. Porque não estamos a lidar com elementos gráficos vetoriais, isto é, de tamanho ajustável em tamanho real, o uso de texturas tal como descrito é apropriado.

Adicionalmente, adiciona-se um ficheiro JSON que mapeia o valor de luminosidade do cinzento num código ISO 3166-1 alpha-3 de países, gerado manualmente e baseado no ficheiro utilizado pela visualização "Small Arms and Ammunition", que continha códigos ISO 3166-1 alpha-2 (duas letras em vez de três). Como os dados tratados inicialmente possuem um código ISO 3166-1 alpha-3 por país, podemos assim associar os dados demográficos a cada nível de cinzento.

Para obter o cinzento selecionado na textura *lookup*, após um clique do rato, utiliza-se um *canvas* e um *raycaster*. O *raycaster* permite obter a interseção da posição do cursor do rato com o *canvas*, e o *canvas* permite associar a localização pressionada a uma localização na textura. Assim, o processo de seleção passa pelos seguintes passos: i) verificar qual o nível de cinzento intersetado pelo *raycaster*; ii) obter o código ISO 3166-1 alpha-3 correspondente ao cinzento intercetado; iii) determinar o país com o código associado; iv) identificar os dados demográficos deste, retirando qualquer cor na textura *select*; e v) pintar de branco a área cinzenta correspondente.

A cor associada a cada país é calculada a partir do valor normalizado do Indicador selecionado. Utiliza-se a biblioteca *chroma.js* para criar um gradiente de cor no modo RGB entre vermelho a verde, onde 0 é vermelho, 1 é verde e qualquer valor entre 0 e 1 é uma cor entre estas. Para obter a cor correspondente ao valor do Indicador, normaliza-se este entre 0 e 1 através da expressão 1 e obtêm-se a cor que identifica este valor normalizado. Na expressão 1, o valor V representa o valor do Indicador i escolhido para um dado País p no AR a .

$$VNormalizado(i, p, a) = \frac{V(i, p, a) - \text{Min}(i, a)}{\text{Max}(i, a) - \text{Min}(i, a)} \quad (1)$$

4.3. Barras Verticais

Para aferir a altura das barras verticais, que deverão representar os valores de um Indicador sem exceder o plano de visão do *browser*, utilizou-se novamente a expressão 1 para normalizar os valores lidos entre 0 e 100 píxeis. Optou-se por uma altura máxima de 100 píxeis por esta ser visualmente confortável em relação à câmara da visualização.

A coloração das barras foi implementada de forma semelhante à coloração dos países (descrito na secção "Globo e Países"). Para este caso, criou-se através da biblioteca *chroma.js* um gradiente de amarelo a azul, onde 0 é amarelo, 1 é azul, e qualquer valor entre 0 e 1 é uma cor entre estas duas cores. Assim, de modo a obter a cor correspondente ao indicador escolhido, normaliza-se entre 0 e 1, através da expressão 1, e obtêm-se a cor mapeada com o valor normalizado.

4.4. Filtragem

Como referido na subsecção anterior, o uso de normalização evita o problema da altura excessiva das barras verticais no plano de visão. Ainda assim, por exemplo, para o Indicador de população total, esta solução leva a que as barras da China e Índia assumam alturas entre 90 e 95 píxeis e que as restantes barras assumam alturas entre 1 e 20 píxeis. No caso da coloração dos países, estes dois países terão cores próximas do verde e amarelo, e os restantes países terão cores avermelhadas e indistinguíveis a olho-nú.

Para poder filtrar os dados de países, utiliza-se a lógica de normalização referida anteriormente mas onde os valores mínimo e máximo são selecionados pelo utilizador. Se o valor for menor que o valor mínimo ou maior que o valor máximo, a barra ou cor no país deverá ser apagada, enquanto que as restantes barras deverão "crescer" ou "decrecer" e as restantes cores deverão progredir mais em direção a uma cor num dos extremos do gradiente.

Assim, a aplicação possui um elemento *slider* no topo da página, associado aos indicadores configurados, e basta ao utilizador escolher os indicadores que pretende analisar e escolher o seu valor mínimo e máximo no *slider*. A representação visual é dinamicamente renderizada à medida que o valor do *slider* é alterado.

A Filtragem de países possibilita a potencial diferenciação de barras e cores para uma melhor leitura dos dados, tal como se observa na figura 4.

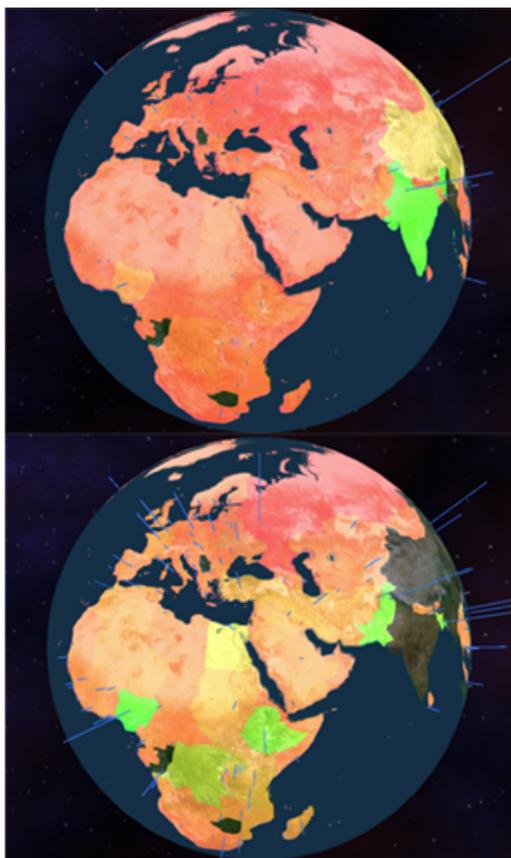


Figura 4. Indicador rácio natalidade-mortalidade mapeado na cor dos países. Quando não-filtrado (em cima) a cor da maioria dos países é uniformemente vermelho e indistinguível entre eles. Quando a China e Índia são filtrados (em baixo), a maioria das cores são distinguíveis e os países com melhor rácio (mais perto do verde) são mais facilmente detetáveis.

Durante o desenvolvimento consideraram-se diferentes formas de representação tri-dimensional de atributos que variam ao longo de um período de tempo, tal como a representação filtrada, onde os dados seriam filtrados de modo a se apresentar apenas um único ano de cada vez, selecionado pelo utilizador e a representação em lápis [CT05]. Esta segunda implicaria a introdução de um objeto adicional por cada país com a forma geométrica de um lápis, onde cada face é colorida desde ponta de carvão (localizado geograficamente no país) até ao cabo deste para representar os valores do primeiro até ao último AR. No entanto, esta representação iria gerar um congestionamento visual, tornando a leitura e comparação entre barras verticais ou entre lápis mais difícil, principalmente para países de pequena dimensão. Assim, optou-se pela representação filtrada, implementando um segundo *slider* horizontal para o utilizador "deslizar" de forma sequencial entre valores de AR.

4.5. Pesquisa

Para implementar a pesquisa, introduziu-se um campo de texto com sugestão de resultados dinâmica que se adapta ao conteúdo escrito no campo. Estas sugestões são adicionadas com a extensão de jQuery "DevBridge Auto-complete"¹⁰, uma extensão onde se especifica qual a lista de dados a dispor nas sugestões e que, por sua vez, os irá dispor automaticamente conforme o que o utilizador escrever no campo de pesquisa. Assim, apresentam-se nas sugestões os nomes de todos os países. Ao selecionar uma das sugestões, obtêm-se o código ISO 3166 alpha-3 e seleciona-se o país correspondente, utilizando o mesmo algoritmo especificado previamente em "Globo e Países".

5. AVALIAÇÃO

A avaliação da visualização implementada permite testar se o modelo concebido acrescenta alguma melhoria em relação às visualizações estado-de-arte no sentido da perceção e da leitura de informação. É necessário perceber se a leitura simultânea de um maior número de indicadores demográficos, representados sobre os Componentes conforme a preferência do utilizador, é intuitiva e facilita a compreensão de relações entre Indicadores.

Um aspeto importante desta fase da avaliação é perceber qual a audiência da visualização, ou seja, qual o contexto social ou profissional em que um dado indivíduo irá utilizar preferencialmente esta visualização em contraste com outras visualizações demográficas tri-dimensionais e até visualizações demográficas bi-dimensionais. Por esta razão, considerou-se que a audiência desta visualização deverá corresponder a dois tipos de utilizador: a) um utilizador com pouca literacia informática, com conhecimento moderado em Ciências Sociais e Estatística; ou b) um utilizador com moderada a alta literacia informática, com conhecimento alto em Ciências Sociais, Estatística ou até Visualização de Dados e Informação.

Tendo em consideração a audiência alvo, utilizaram-se dois tipos de avaliação para cobrir as diferentes necessidades e níveis de especialização dos utilizadores.

5.1. Teste de usabilidade

A primeira avaliação correspondeu a um teste de usabilidade, em que se pediu a dois utilizadores com pouca literacia informática, com idades de 23 e 59 anos e sem conhecimento nem experiência no domínio da Visualização de Informação para realizar um conjunto de tarefas e fornecer algumas observações finais sobre a facilidade de utilização da visualização e resolução destas tarefas. Cada indivíduo foi observado em separado e cada teste durou aproximadamente 7 minutos.

A primeira fase do teste, com duração entre 30 segundos a 1 minuto, constituiu numa introdução breve sobre as funcionalidade básicas da aplicação, nomeadamente a possibilidade de seleção e pesquisa de países, os Componentes visuais, a personalização de Indicadores e a Filtragem.

¹⁰<https://www.devbridge.com/sourcery/components/jquery-autocomplete/>

A segunda fase do teste, com duração entre 5 a 6 minutos, correspondeu à resolução de um conjunto de cinco tarefas. Todas as tarefas incidiam na resolução de um dado problema, sendo por isso não só quanto à precisão da resposta ao problema como também quanto à rapidez na execução da tarefa e quanto à incidência de falsos positivos, onde o utilizador assume que concluiu a tarefa corretamente quando esta não o foi. As cinco tarefas corresponderam à resolução das seguintes questões, introduzidas pela mesma ordem ao utilizador:

1. Qual é a população total de Portugal?
2. Qual é o crescimento populacional de Porto-Novo?
3. Sem utilizar a seleção ou pesquisa, qual é o país com maior natalidade?
4. Sem utilizar a seleção ou pesquisa, entre o Brasil e a Argentina, qual o que apresenta uma maior razão de nados-vivos / óbitos?
5. Sem utilizar a seleção ou pesquisa e associando o Indicador "Taxa Bruta de Mortalidade" à cor dos países, qual o país com a maior taxa de mortalidade?

O objetivo principal das primeiras três tarefas, com um grau de dificuldade baixo, é colocar os utilizadores mais à vontade com a aplicação e incentivá-los a utilizar as funcionalidades, como a pesquisa. O objetivo das duas últimas tarefas, com um grau de dificuldade maior, é introduzir desafios para avaliar se estas funcionalidades são fáceis de utilizar para chegar às soluções.

É de salientar que a quinta tarefa tem como principal objetivo perceber se o uso das cores vermelho e verde para representar respetivamente valores baixos a valores altos de mortalidade iria confundir o utilizador na eventualidade de este assumir o significado das cores respetivamente pela sua conotação habitual negativa e positiva. Ou seja, o utilizador poderá inferir que um país com cor verde tem valores "melhores", logo irá ter um menor valor de mortalidade (e vice versa), e gerando um falso-positivo.

Os resultados dos testes realizados são apresentados na tabela 1, sendo: 'Tempo', o tempo que o utilizador utilizou para concluir a tarefa; 'Ajuda', se o utilizador solicitou ajuda; 'Confiança', se o utilizador tem confiança na resposta dada; e 'Correção', se o utilizador chegou ao resultado correto.

A última fase do teste, com duração de 1 minuto, passou por inquirir os utilizadores sobre que aspetos da visualização poderiam ter sido melhorados e que tarefas consideraram mais fáceis de executar. As observações obtidas indicam que a visualização foi fácil de utilizar e que a funcionalidade de Filtragem é extremamente útil para obter informação gráfica de certos países com mais detalhe. O segundo utilizador, que respondeu erradamente e com confiança à quinta tarefa, afirmou que momentaneamente não se recordou da escala de cores de vermelho a verde

Tabela 1. Resultados dos testes de usabilidade realizados.

		Utilizador 1	Utilizador 2
Tarefa 1	Tempo	2 s	8 s
	Ajuda	Não	Não
	Confiança	Sim	Sim
	Correção	Sim	Sim
Tarefa 2	Tempo	21 s	16 s
	Ajuda	Não	Não
	Confiança	Sim	Sim
	Correção	Sim	Sim
Tarefa 3	Tempo	11 s	47 s
	Ajuda	Não	Não
	Confiança	Sim	Sim
	Correção	Sim	Sim
Tarefa 4	Tempo	33 s	39 s
	Ajuda	Não	Não
	Confiança	Sim	Sim
	Correção	Sim	Sim
Tarefa 5	Tempo	26 s	43 s
	Ajuda	Não	Não
	Confiança	Sim	Sim
	Correção	Sim	Não

usada, mas em vez disso da conotação negativa do vermelho. Ambos os inquiridos classificaram com 4 (numa escala do tipo Likert de 1 a 5) a satisfação de utilização da aplicação.

5.2. Avaliação Heurística

Foi também realizada uma avaliação heurística por dois analistas com conhecimentos no domínio de Visualização de Informação utilizando heurísticas apropriadas e à escolha do avaliador. Ambos os avaliadores utilizaram as heurísticas de Nielsen [Nielsen 90], dado o seu potencial para encontrar problemas de usabilidade [Nielsen 92]. As avaliações foram feitas em separado, e cada avaliador elaborou um relatório onde indicou os potenciais problemas de usabilidade, as heurísticas não cumpridas, bem como a gravidade de cada problema, indicando também aspetos em que as heurísticas eram cumpridas.

De um modo geral, a visualização destacou-se positivamente nas heurísticas *Correspondência entre o sistema e o mundo real*, *Controlo e livre-arbítrio do utilizador*, *Consistência e Adesão a Normas*, *Reconhecimento em vez de lembrança*, *Flexibilidade e Eficiência* e *Desenho de ecrã estético e minimalista*. Isto significa que a representação gráfica de dados demográficos com escolha livre de Componentes parece não apresentar problemas severos de usabilidade. No entanto, ambos os avaliadores encontraram um aspeto negativo correspondendo ao não cumprimento das heurísticas *Visibilidade do estado do sistema* e *Ajuda e Documentação*, relacionado com a falta de documentação ou instruções que indiquem ao utilizador os próximos passos a tomar quando a visualização é iniciada.

6. CONCLUSÕES E TRABALHO FUTURO

Neste artigo propõe-se uma visualização de dados demográficos tridimensional sobre um globo e descreve-se a implementação dum protótipo que permite a exploração interativa desses dados de forma mais completa e flexível do que as aplicações com objetivos análogos atualmente disponíveis. A usabilidade deste protótipo foi avaliada através de avaliação heurística e testes de usabilidade. Um dos aspetos mais importantes desta visualização é a filtragem dos dados que facilita o processo de análise. Embora, os resultados dos testes de usabilidade pareçam indicar que a funcionalidade de filtragem não é evidente para os utilizadores, uma vez aprendida, é utilizada com frequência para facilitar a visualização de países com valores mais pequenos dos indicadores em estudo. Outro resultado da avaliação aponta para que a possibilidade dada aos utilizadores de associar cada tipo de indicador a uma forma de representação visual (por exemplo, cor), pode conduzir a associações menos intuitivas e provocar erros de interpretação. Por esta razão, propõe-se como trabalho futuro, facultar aos utilizadores a possibilidade de escolher não apenas a associação de indicadores a formas de representação visual (Componentes), mas também as escalas de cor utilizadas. É ainda de salientar que os utilizadores recorreram à pesquisa e seleção sem solicitar ajuda, o que sugere que estas funcionalidades são intuitivas e permitem realizar facilmente um *zoom* dos dados. Também os resultados da avaliação heurística foram geralmente positivos.

Em geral, o modelo proposto itera sobre os modelos estado-de-arte analisados e adiciona componentes que tornam a exploração visual e interativa de dados demográficos mais fácil e intuitiva.

Os conceitos e funcionalidade do modelo proposto neste artigo podem ser utilizados em contextos para além do domínio de estudo demográfico. Sendo assim, consideramos o trabalho atual como um ponto de partida para o desenho de uma arquitetura genérica que recebe dados, sob a forma de uma API (Application Program Interface) estruturada, e os representa no globo. Deste modo, a funcionalidade de seleção do país, seleção do ano de recolha, pesquisa, filtragem e navegação do globo estariam já implementadas e abstraídas, agilizando o processo de apresentação de dados num globo por parte de um desenvolvedor em qualquer página web. Esta API poderia também incluir ou esconder funcionalidades conforme a composição dos dados, por exemplo, se os dados não têm uma data de recolha, então o *slider* de AR é escondido.

7. AGRADECIMENTOS

Este trabalho é suportado por Fundos Nacionais através da FCT - Fundação para a Ciência e a Tecnologia, no contexto dos projetos UID/CEC/00127/2013 e Incentivo/EEI/UI0127/2014.

Os autores gostariam de agradecer a Luís Silva, Bruno Garcia, Pedro Miguel, Isabel Nascimento, Rui Simões e João Pedrosa pelos valiosos contributos que deram no teste e melhoria da aplicação.

Referências

- [Arts 11] Google Data Arts. World population. <http://data-arts.appspot.com/globe>, 2011.
- [Arts 14] Google Data Arts. Small arms and ammunition - imports & exports. <http://armsglobe.chromeexperiments.com>, 2014.
- [Chi 98] Ed H. Chi. An operator framework for information visualization systems. *Information Visualization Symposium*, 1998.
- [CT05] P. Schulze-Wollgast; H.Schumann C. Tominski. 3d information visualization for time dependent data on maps. *Proceedings of the International Conference on Information Visualisation*, 2005.
- [Dirksen 12] Jos Dirksen. World population density - 2010. <http://www.smartjava.org/examples/population>, 2012.
- [Mazza 09] Riccardo Mazza. Introduction to information visualization. *Springer-Verlag London*, 1, 2009.
- [Nielsen 90] Jakob Nielsen. Heuristic evaluation of user interfaces. *Proceedings of the ACM CHI'90 Conference*, páginas 249–256, 1990.
- [Nielsen 92] Jakob Nielsen. Finding usability problems through heuristic evaluation. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, páginas 373–380, 1992.
- [Pridmore 11] Ralph W. Pridmore. Complementary colors theory of colorvision: Physiology, color mixture, colorconstancy and color perception. *Color. Res. Appl.*, 2011.
- [Zhifang 13] Wang Zhifang. China and us population. <http://newnaw.com/pub/js/webglglobe/chinaanduspop>, 2013.



Virtual, Mixed and Augmented Reality

Contents

How to Deal with Motion Sickness in Virtual Reality <i>Bruno Patrão, Samuel Pedro, Paulo Menezes</i>	40
Incremental Reconstruction Approach for Telepresence or AR Applications <i>Luís Almeida, Paulo Menezes, Jorge Dias</i>	47
Modelo de descrição de experiências multissensoriais multiutilizador <i>Rui Rodrigues, Eduardo Magalhães, Hugo da Silva, Maximino Bessa, António Coelho, Augusto Sousa</i> 55	
Interactive configurable virtual environment with Kinect navigation and interaction <i>João Pinto, Paulo Dias, Sérgio Eliseu, Beatriz Sousa Santos</i>	63
Visualização de Objetos Off-Screen em Realidade Aumentada Móvel <i>Gonçalo Silva, Maria Beatriz Carmo, Ana Paula Afonso, Ana Paula Cláudio, António Ferreira</i>	71

How to Deal with Motion Sickness in Virtual Reality

Bruno Patrão Samuel Pedro Paulo Menezes

Institute of Systems and Robotics
Department of Electrical and Computer Engineering
University of Coimbra
3030-290 Coimbra, Portugal
{bpatrao, samuel, paulo}@isr.uc.pt

Abstract

In this paper we present a exploratory study on the physiological responses when experiencing motion sickness in Virtual Reality (VR). To this end, we developed a VR application that can induce motion sickness. Using it, an experiment was performed where a group of users were subject to different types of observable motions, and for each the reported sensations together with a set of bio-signals were registered. The analysis of the collected results enable us to establish a relationship between VR/Motion sickness and the principal elements that may cause it, as well as the existence of some correlation between the discomfort felt by the participants and detectable changes in measurable physiological data. These results can serve both as a guide to designing VR-based applications, complementing the existing ones, and to enable the development of automatically adaptable ones preventing or reducing the discomfort for the users of this type of technology.

Keywords

Motion Sickness, Virtual Reality, Bio-Signals, Usability.

1 INTRODUCTION

With the emergence of Virtual Reality (VR) into the more commercial mainstream, issues have begun to arise in relation to the wellbeing and comfort of its users. Recent studies have shown that the biggest barrier of using VR systems is the so called VR sickness, pointing that this kind of sickness may be limiting the effective use of training, rehabilitation or gaming tools in VR [Barrett 04, Brooks 10, Wiederhold 14]. Although there is still limited information about the related physiological mechanisms, there is evidence that it is a form of motion sickness caused essentially by the mismatch between the motion perceived visually and through the vestibular system [Reason 75]. In VR, it happens mainly when the visual system detects apparent motion, whereas the vestibular system informs the brain that no movement is occurring and that the body is standing still. This effect can also occur in situations where the perceived movement is caused by the user head movements. Apparently this should contradict the "mismatch rule", but there are at least two situations where it may occur: 1) low/variable screen update rate - this frequently happens due to high computational load of the scene rendering or due to limitations of the head mount display (HMD) hardware. The low/variable update

frequency of the visualised scene, makes the human brain detect "jumps" instead of perceiving a continuous motion as reported by the vestibular system. 2) high difference between the head rotation movements and the perceived motion of the objects. This is similar to the discomfort caused by new glasses where a change on their graduation (refraction), or the use of progressive lenses requires an adaptation period.

1.1 Motion Sickness and the Autonomic Nervous System

There are morphological and physiological connections between the vestibular and autonomic nervous system, including its sympathetic and parasympathetic divisions. Motion sickness, which is produced by conflicting inputs from visual, vestibular and somatosensory afferents, generally carries vestibule-autonomic responses in humans. Sympathetic (SNA) and parasympathetic nervous activity (PNA) contribute to those responses, such as, the heart rate variability, skin conductance variability or body temperature changes. While the PNA controls the body's response in rest, SNA is responsible for the internal response in body's fight-or-flight, meaning the reaction to a stimuli [Hu 91, Farmer 14].

1.2 Contribution and Structure of the Paper

The contributions of this paper are two-fold: it analyses the most common elements that are responsible to induce motion sickness in VR applications, and establishes a relationship between some motion sickness symptoms and measurable physiological responses. Both may contribute to improve the design of future VR applications aiming at reducing the discomfort that its use frequently induces in users.

The remaining of the paper is as follows: section 2 presents the experimental setup, describing both the built immersive application and the acquired physiological signals. Section 3, presents the experimental protocol and evaluation procedure. Section 4 shows the obtained results and analyses them. Section 5 concludes and presents future directions.

2 DEVELOPED SYSTEM

In order to achieve our goal we developed a set of scenarios, taking into account different assumptions for each of them. It is known that in VR, motion sickness occurs when there are asynchronous events between our inertial and visual system. In this case our visual system perceives the movement but our body does not.

2.1 Virtual Environment

The VR system developed for this study consists in a virtual environment presented to the user through a Head-Mounted Display (HDM) which tracks all head movements. The user is immersed and can freely look around in the virtual environment.

The virtual environment is a street with houses and the user moves along a clockwise circle trajectory around it. During the experiment different combinations of involuntary camera behaviours and virtual elements are presented to the participant in order to induce or reduce the (un)desire sickness. Let's define voluntary movements as all the camera movement that the user performs voluntarily and involuntary movements as the random camera behaviour not controlled by the user. Figure 1 shows the involuntary movements that were tested.

The virtual element used as frame reference is a car cockpit. Figure 2 shows the point of view of the user without and with the virtual cockpit.

2.2 Physiological Data

Electrocardiography

Electrocardiography (ECG) is an exam able to register the variation in potential created by the heart beating. It can be used to assess the Heart Rate Variability (HRV) which is the study of the variations associated with the Heart Rate (HR).

The Heart Rate measures how many times the cardiac muscle contracts and relaxes per unit of time, triggering depolarisation and subsequent polarisation of the myocardial cells.

Camera Behaviour	Description
	Eight Shape
Motion Behaviour	
	Constant Velocity
	Accelerations

Figure 1: Involuntary camera and motion behaviour presented to the user.

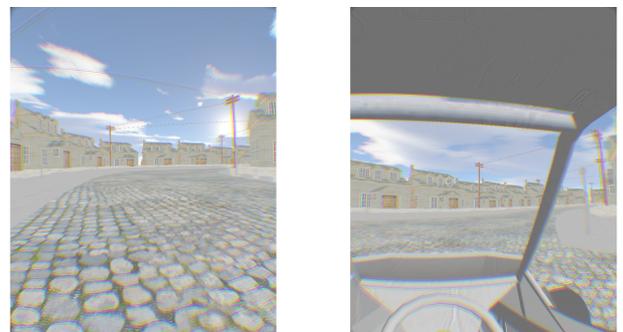


Figure 2: User point of view without cockpit (left) and with cockpit (right).

An ECG sample signal is better-known by its well defined QRS complex, which is the denomination given to the succession of the three deflections of a typical ECG wave-form (figure 3). The instant Heart Rate can be calculated by inverting the distance between R-peaks and multiplying it by 60. In situations of mental or physical arousal, the heart rate shall increase [Edhouse 02].

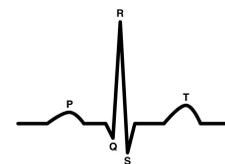


Figure 3: Illustration of the wave-form of an ECG signal.

Heart rate variability (HRV) is the consequence of the iteration of both sympathetic and parasympathetic halves of the automatic nervous system. The parasympathetic system shall slow the Heart Rate whereas the sympathetic accelerates it [Acharya 06].

Electrodermal Activity

Electrodermal activity (EDA), also known as Galvanic skin response, is related to the wet level of the sweat glands. These are controlled by the sympathetic system,

varying the electrical skin conductance, specially once an individual is presented a stimuli [Moore 03].

This measure is an indicator of the level of the person's arousal. EDA provides a reliable measure to assess the automatic nerve system function, providing an easy tool to get a person's internal state.

Physically, Galvanic skin response is often shown as a modification of the electrical features of the skin in response to a definite stimuli, capturing the variation of electrical potential from the skin surface. In fact, the increasing of EDA level depends more on the surprise effect of the stimuli than its strength [Ahuja 03, Tarvainen 01]. It is also known that EDA signal has the tendency to decrease once the person shows habituation [Hugdahl 96].

Body Temperature

Human body temperature varies depending on the place where it is measured. In this experiment the body temperature is measured upon the skin on the wrist. Nonetheless, the temperature tends to be lower in the surface than when measured internally, as an example below the tongue.

When a person is in some physical or mental stress, this measure tends to increase [Longo 11]. An increase in body temperature can be achieved for example by reduced convective heat loss via sympathetically controlled cutaneous vasoconstriction. Nevertheless, sweating can influence this measure. Sweating is a way our body uses to regulate our surface body temperature [Illigens 09].

Respiration Rate

The respiratory system has as primary function to supply oxygen to the blood in order to be delivered to all parts of the body. This is achieved through breathing. When breathing, we inhale oxygen and exhale carbon dioxide. This process involves the movement of the diaphragm which consequently is expressed in a movement of expanding and compressing of the rib cage.

Using an flex sensor, one can measure the amplitude and rate in which this process is carried by the person. This can be a good indicator of the person's internal state. It is known that a controlled rate of breathing can be a mechanism to control nausea, one of the symptoms of motion sickness [Farmer 14].

Body Acceleration

Accelerometers are inertial sensors used to measure three-dimensional accelerations. When placed on a moving body, it's capable of accurately sense it's movements.

In this case the two used accelerometers are attached one to the person's wrist so that he/she can signal simulation sickness episodes and the other one to the head of the person to trace motion artefact implied in the respiration signal.

2.3 Questionnaire

To assess the subjective level of the motion sickness symptoms we used the Motion Sickness Assessment Questionnaire (MSAQ). The participants were asked to rate how accurately the following statements describe their experience.

The scale goes from 1 to 9, where 1 means that the statement does not apply at all and 9 it severely occur.

1. I felt sick to my stomach (G)
2. I felt faint-like (C)
3. I felt annoyed/irritated (S)
4. I felt sweaty (P)
5. I felt queasy (G)
6. I felt lightheaded (C)
7. I felt drowsy (S)
8. I felt clammy/cold sweat (P)
9. I felt disoriented (C)
10. I felt tired/fatigued (S)
11. I felt nauseated (G)
12. I felt hot/warm (P)
13. I felt dizzy (C)
14. I felt like I was spinning (C)
15. I felt as if I may vomit (G)
16. I felt uneasy (S)

Note. G: Gastrointestinal; C: Central; P: Peripheral; S: Sopite-related.

The overall motion sickness score is obtained by calculating the percentage of total points scored: (sum of points from all items/144) x 100. Subscale scores are obtained by calculating the percent of points scored within each factor: (sum of gastrointestinal items/36) x 100; (sum of central items/45) x 100; (sum of peripheral items/27) x 100; (sum of sopite-related items/36) x 100.

3 EXPERIMENTAL PROTOCOL AND EVALUATION

3.1 Participants

Participants were eight volunteers (five male and three female, mean age of 25.75 and standard deviation of 2.49 years old). All participants reported healthy and normal or corrected to normal vision. Five had experience with video games and three of them had previous experience in immersive VR, but none had prior knowledge of the current experiment. All research ethical principles were attained.

3.2 Method (Procedure)

The experiment starts with the participant resting for a period of two minutes, followed by six minutes of VR exposure and it ends with a period of two minutes of rest, as illustrated in figure 4. Furthermore, participant's physiological data were collected during the full ten minutes of the experiment.

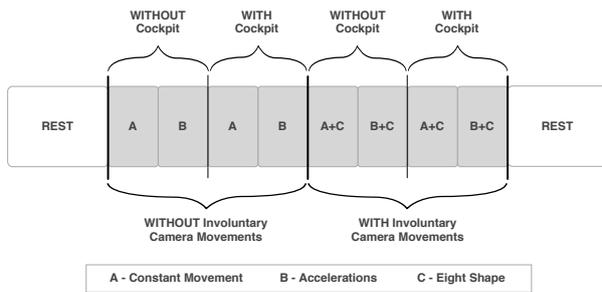


Figure 4: Experiment timeline.

At the end, participants were asked to remember and rate the best (as +1) and the worst (as -1) combination of camera behaviours and virtual elements during the VR exposure and fill a questionnaire (MSAQ) based on the worst experienced situation.

3.3 Measurements and Questionnaire

3.3.1 Physiological Signal Assessment

Willing to assess motion sickness, the signals acquired from the worn sensors must be preprocessed to remove any noise and analysed to extract some selected features. As the behaviour we want to assess has implicitly some duration in time, we perform analysis over samples that represent a sliding window of 45 seconds of activity with an overlap of 15 seconds.

Signal Preprocessing

Signal preprocessing can easily be divided into two main domains.

Signal filtering (Butterworth filter)

Signal filtering is used in this case to limit the bandwidth of frequencies allowed. A Butterworth Low pass filter is used to remove the noisy high frequency domain. In the same sense, some signals were filtered with a Highpass filter. This step is only used in the ECG signal as we can assume a tendency to zero, allowing the removal of the effect of baseline wander. This is important for a correct R-peaks detection, thus computing a reliable Heart Rate measure.

Baseline subtraction

A baseline is in this case defined as the value to which the signals will tend in a state of rest. Some of the signals, for instance EDA and Heart Rate, have a great interpersonal variation. Knowing this, one needs to remove this varying variable from the data. In this way, we will consider the rest state of the person as a zero baseline in the signal.

Features Extracted

In order to best capture the sense of the data, a set of features were calculated from the data window. These features are intended to best summarise the person's state

in the window moment. Below we present selected metrics used from each of the bio-signals.

ECG features

In order to calculate ECG features, one has to detect R-peaks in signal. To better understand the technique, please refer to [Pedro 13]. Having detected the heart beats, one can compute the NN intervals, which is defined by the difference between successive normal R-peaks.

Heart Rate is defined by the number of times the heart beats per unit of time. It can be calculated by $\frac{1}{NN} \times 60$. This measure helps in the understanding of the anxiety and stress in the person.

RMSSD is defined by the Root mean square of the mean of successive NNs,

$$\sqrt{\frac{\sum_{i=1}^{n-1} D_i^2}{n-1}} \quad (1)$$

This measure is known to be an accepted measure of parasympathetic activity [Sztajzel 04].

LF/HF ratio is the ratio of low-high frequency power. To calculate this measure it is necessary to analyse the frequency domain of the ECG signal, dividing the low frequency power by the high frequencies. It reflects the global sympatho-vagal balance (which is the balance between sympathetic and parasympathetic activity) and will be used as a measure of this balance [Sztajzel 04]. As it is known, the vagus nerve complex is related to the major reported motion sickness symptoms (e.g. nausea, disorientation and stomach awareness)[Hettinger 03].

EDA features

The number of peaks (nPEAKS) in the temporal window in analysis is used to analyse which period of time was most stressful for the user.

MAXMIN is the difference between the highest value of the window and the lower can assess the degree of change inside one window. If the maximum appear before the minimum the computation is a negative difference.

Mean The mean value of the window assesses the person's state over time.

The MEAN and the MAXMIN are extracted from both EDA and Temperature signals.

Respiration features

Rate or Frequency is used to analyse over time variation. Assuming breathing can be approximated to a sine function inside the temporal window, using FFT transformation, one can calculate the preponderant frequency in the spectrum. The variation of this measure can be related to the Heart Rate variation and so the autonomic nervous system activity.

4 RESULTS AND DISCUSSION

This section presents the results from both the bio-signal analysis and the subjects' questionnaire answers. We start by analysing the bio-signals, correlating those results with the participants subjective opinion of the best and worst reported scenarios.

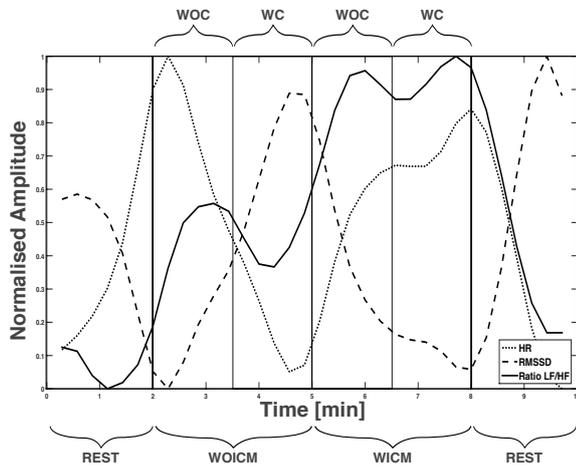


Figure 5: ECG results.

Figure 5 presents the result of the three ECG monitored features over time in the experiment. The measures are the mean values from all the subjects and are normalized to allow simultaneous illustration. One can start by noting the similarity of the measures tendency in the rest periods. Both *Heart Rate* (dotted line) and *LF/HF ratio* (dashed line) return to baseline value in the end of the experiment. In fact, *RMSSD* (solid line) value show an even higher parasympathetic activity (rest control) in the end of the experiment.

Being the first and the second scenarios (*WOICM*) evaluated as the most comfortable by the users (figure 7), one can see lower values of *LF/HF ratio* and higher *RMSSD* values, indicating a predominant activity of the parasympathetic system. Heart Rate rise in the first scenario can be explained by the subject's surprise and expectation in the beginning of the experiment. The worst reported scenario was the fourth (*WC & WICM*) showing higher *LF/HF ratio* and Heart Rate values as expected while low *RMSSD* value.

It is important to note a particular scenarios transition. The trend change in the three measures in the transition between scenario two (one of the best) and three (one of the worst), shows that the user's subjective opinion about the scenarios can be confirmed by the bio-signal analysis.

The illustration 6 presents the mean results for EDA features. It can be initially noted that values in both rest periods tend to some low value. Despite this, the mean value of EDA (solid line) starts higher, reason being for the initial enthusiasm of the subject before the experiment. Nonetheless, this value gets lower throughout the initial

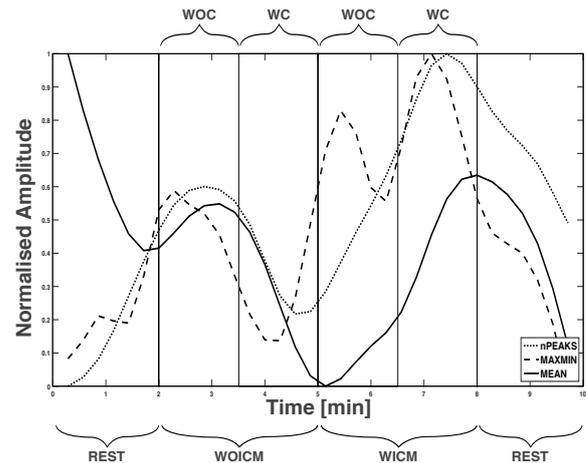


Figure 6: EDA results.

rest period.

One can see that all the values get higher in the beginning of the first scenario, due to the surprise effect. As it was stated in section 2.2, the effect gets weaker with time (stimuli habituation) and so, if we presented the same scenario to the user throughout all the experiment, one could expect a first EDA rise by the stimuli and a return to the rest state afterwards. As it can be seen, when the person enters the worst reported scenarios, all the measures rise, due to motion sickness discomfort. The number of peaks (dotted line) can be interpreted as sweating spikes, while the *MAXMIN* values (dashed line) show how severe these rises get. Moreover, the mean EDA values suggest the robustness of the previous two measures.

For respiration and temperature features, there were no significant changes over the experiment.

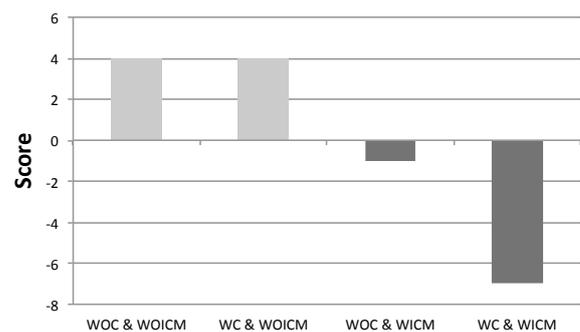


Figure 7: Best (light gray) and worst (dark gray) reported combinations.

Figure 8 presents the overall average score for motion sickness level before and after the experiment for the worst reported scenario. It is assumed that since the participants did not manifest any prior statements of motion sickness at the beginning, the score level starts at zero. At the end of the experiment, the overall average score rose to 36.02% with a standard deviation of 14.70%.

Finally, we present the results of the questionnaire's

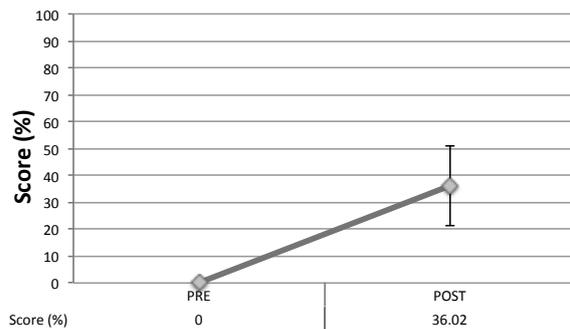


Figure 8: Overall average score of MSAQ.

answer into four different motion sickness sub-scales. In fact, motion sickness may be better quantified as a multidimensional scales, dividing it into different type of symptoms. The advantage of separating them is that the same overall score for two participants can be achieved by having different sub-scales scores. Figure 9 presents the average score for each of these four components. It is noticeable that the Central and Peripheral sub-scales, most related to Heart Rate and EDA measures, got higher scores.

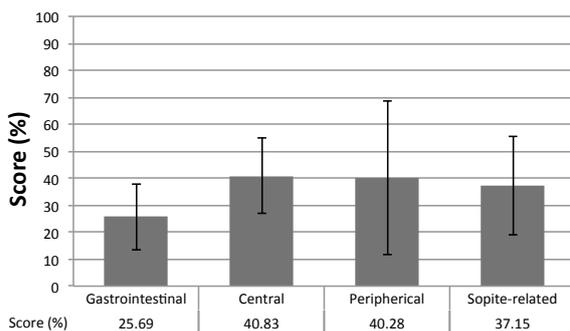


Figure 9: Sub-scales average score of MSAQ.

From the experiment results and observations, we can present some assumptions on how to induce or reduce motion sickness in a VR environment.

How to induce motion sickness

- Changing the head orientation without the user input;
- Modifying the field of view or zooming in and out;
- Variations in acceleration or speeding up and down.

How to reduce motion sickness

- Maintaining the immersion from the start until the very end of the experiment. There must be always a response to the user input (in loading screens, transitions, etc.), if the screen stops it give the sensation that the system is not responding, removing the immersion;
- Adding a frame reference also helps to prevent the user to feel unwell, as long as there are no involuntary

camera movements (the user must be always on control of the camera). These kind of elements seems to work great because the user have a reference that connects him/her to the virtual environment. For example, a cockpit or a windshield that is with the user all the time in his/her peripheral vision work very well for this purpose;

- VR must be treated as theatre not as a film, so the user's head is someone eyes not a camera. Avoid camera animations and tricks, such as, zoom in and out, these can lead to the undesired sickness.

5 CONCLUSION

We presented a exploratory study on the physiological responses when experiencing motion sickness in Virtual Reality (VR). A VR application was developed to induce motion sickness. Using it, an experiment was performed where a group of users were subject to different types of observable motions, and for each the reported sensations together with a set of bio-signals were registered. The analysis of the collected results enabled us to establish a relationship between VR/Motion sickness and the principal elements that may cause it, as well as the existence of some correlation between the discomfort felt by the participants and detectable changes in measurable physiological data. These results can serve both as a guide to designing VR-based applications, complementing the existing ones, and to enable the development of automatically adaptable ones preventing or reducing the discomfort for the users of this type of technology.

References

- [Acharya 06] U. Rajendra Acharya, K. Paul Joseph, N. Kannathal, Choo Min Lim, and Jasjit S. Suri. Heart rate variability: A review. *Medical and Biological Engineering and Computing*, 44(12):1031–1051, 2006.
- [Ahuja 03] N.D. Ahuja, a.K. Agarwal, N.M. Mahajan, N.H. Mehta, and H.N. Kapadia. GSR and HRV: its application in clinical diagnosis. In *16th IEEE Symposium Computer-Based Medical Systems, 2003. Proceedings.*, 2003.
- [Barrett 04] Judy Barrett. Side effects of virtual environments: A review of the literature. Technical report, DTIC Document, 2004.
- [Brooks 10] Johnell O Brooks, Richard R Goodenough, Matthew C Crisler, Nathan D Klein, Rebecca L Alley, Beatrice L Koon, William C Logan, Jennifer H Ogle, Richard A Tyrrell,

- and Rebekkah F Wills. Simulator sickness during driving simulation studies. *Accident Analysis & Prevention*, 42(3):788–796, 2010.
- [Edhouse 02] June Edhouse, William J Brady, and Francis Morris. *ABC of clinical electrocardiography: Acute myocardial infarction-Part II.*, volume 324. BMJ Books, 2002.
- [Farmer 14] Adam D. Farmer, Yasser Al Omran, Qasim Aziz, and Paul L. Andrews. The role of the parasympathetic nervous system in visually induced motion sickness: Systematic review and meta-analysis. *Experimental Brain Research*, 232(8):2665–2673, 2014.
- [Hettinger 03] Lawrence J Hettinger and Michael W Haas. *Virtual and adaptive environments: Applications, implications, and human performance issues.* CRC Press, 2003.
- [Hu 91] Senqi Hu, Wanda F Grant, Robert M Stern, and Kenneth L Koch. Motion sickness severity and physiological correlates during repeated exposures to a rotating optokinetic drum. *Aviation, space, and environmental medicine*, 1991.
- [Hugdahl 96] K Hugdahl. Cognitive influences on human autonomic nervous system function. *Current opinion in neurobiology*, 6(2):252–258, 1996.
- [Illigens 09] B. M W Illigens and Christopher H. Gibbons. Sweat testing to evaluate autonomic function. *Clinical Autonomic Research*, 19(2):79–87, 2009.
- [Longo 11] D Longo and A Fauci. *Harrison’s Principles of Internal Medicine.* New York: McGraw-Hill., 2011.
- [Moore 03] Melody M Moore and Umang Dua. A galvanic skin response interface for people with severe motor disabilities. *ACM SIGACCESS Accessibility and Computing*, (77-78):48–54, 2003.
- [Pedro 13] Samuel Pedro. Sensor-based Detection of Alzheimer’s Disease-Related Behaviours. Master’s thesis, University of Coimbra, 2013.
- [Reason 75] James T Reason and Joseph John Brand. *Motion sickness.* Academic press, 1975.
- [Sztajzel 04] Juan Sztajzel et al. Heart rate variability: a noninvasive electrocardiographic method to measure the autonomic nervous system. *Swiss medical weekly*, 134:514–522, 2004.
- [Tarvainen 01] Mika P. Tarvainen, Anu S. Koistinen, Minna Valkonen-Korhonen, Juhani Partanen, and Pasi a. Karjalainen. Analysis of galvanic skin responses with principal components and clustering techniques. *IEEE Transactions on Biomedical Engineering*, 48(10):1071–1079, October 2001.
- [Wiederhold 14] Brenda K Wiederhold and Stéphane Bouchard. Sickness in virtual reality. In *Advances in Virtual Reality and Anxiety Disorders*, pages 35–62. Springer, 2014.

Incremental Reconstruction Approach for Telepresence or AR Applications

Luis Almeida
ISR, Univ. Coimbra
Polytechnic of Tomar
Tomar, Portugal
laa@ipt.pt

Paulo Menezes
ISR, Univ. Coimbra
Coimbra
Portugal
paulo@isr.uc.pt

Jorge Dias
ISR, Univ. Coimbra
Coimbra
Portugal
jorge@deec.uc.pt

Abstract

This paper proposes an on-line incremental 3D reconstruction framework aimed at fulfilling the needs of telepresence or human machine interaction applications. The research presents a teleconference system that improves and induces the feeling that persons are in the presence of each other. A free viewpoint method, based on realistic user's appearances, is proposed to simulate a real face-to-face meeting. The contributions are: a new incremental version of Crust algorithm that enables incremental fusion of sensor data and a confidence-based method that automatically decides whether or not to integrate newly acquired data in the existing model based on measure uncertainty and novelty. To avoid the classical stereo vision reconstruction problems, the method bases on hybrid sensors to acquire simultaneous depth information and the corresponding texture image (e.g. kinect). This enables the alignment between acquired data and pre-acquired model by maximizing a criterion that is related with the matching between visual features and between acquired shapes. A mesh based representation enables the use of the surface topological geometric information during the data model integration process.

Keywords

Three-Dimensional Graphics and Realism, Augmented Reality, Reconstruction, Range Data, Tracking, Telepresence

1 INTRODUCTION

Widely used video teleconference applications (ex: Cisco WebEx, Citrix GoToMeeting, Microsoft Skype, Google Hangouts or Apple Facetime) are not replicating important real face-to-face meeting cues, like eye-to-eye contact establishment, gesture reconnaissance, body language or facial expressions. Nevertheless, recent advances on sensing, display and computation technology are creating the ideal condition for affordable consumer 3D applications in Augmented Reality (AR), Virtual Reality (VR) or Human Machine Interactions (HMI). Our application concept goal is depicted in Figure 1, where user's locations setup, ideally equipped with displays, video cameras, depth sensor, microphones and speakers, enables users to communicate and interact remotely experiencing the benefits of a face-to-face meeting in full size. It includes a 3D capture, reconstruction and virtual view synthesis display system.

There are some notable works that realistically exploit the user's appearance for tele-immersion like those developed at UC Berkeley [Kurillo 08] and at GrImage at INRIA [Petit 09]. Both use video cameras array to perform real-time full body 3D reconstructions leading to some weaknesses, like: reconstruction problems due to the lack of accuracy in low-texture or repeated pattern re-

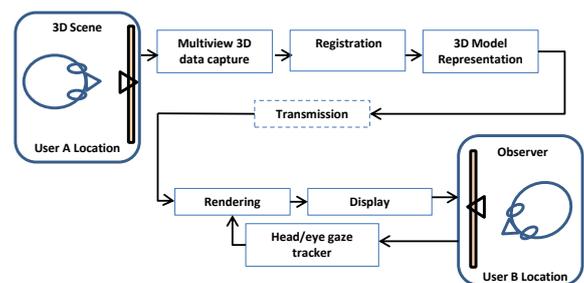


Figure 1. Face to face meeting through technology mediation, line of sight preserving method. Overview of the reconstruction algorithm that aims to continuously generate a realistic body model, transfer the model and reconstruct on a remote common display or virtual environment according, each user's viewpoint by a tracking process. The proposed real-time 3D full reconstruction system combines visual features and shape-based alignment between consecutive point clouds while the mesh model representation is updated incrementally using a new Crust based algorithm.

gions, high cost acquisition data setups, high power computational requirements, and their unsuitability for domestic use. Recent RGB-D reconstruction related works are using alignment and integration approaches based on SLAM sparse methods [Beck 13][Almeida 13]. Henry et al. [Henry 12] combine visual feature matching with ICP-based pose estimation to build a pose-graph which they optimize to create a globally consistent map. Newcombe et al. [Newcombe 11] presented an improved accurate solution known as KinectFusion which uses a new algorithm for real-time dense 3D mapping. KinectFusion integrates depth maps from the Kinect into a "truncated signed distance function" (TSDF) representation. The required alignment to fuse the depth maps is based on the iterative closest point algorithm (ICP), that runs on a GPU for obtaining real time performance.

Our contribution is a real-time 3D full reconstruction system that combines visual features and shape-based alignment between consecutive point clouds while the mesh model representation is updated incrementally using a new Crust based algorithm.

The paper is organized as follows. Section 2 describes the proposed reconstruction methodology, section 3 presents some experimental results and discussion and, section 4 presents the future work and conclusions.

2 MESH GENERATION

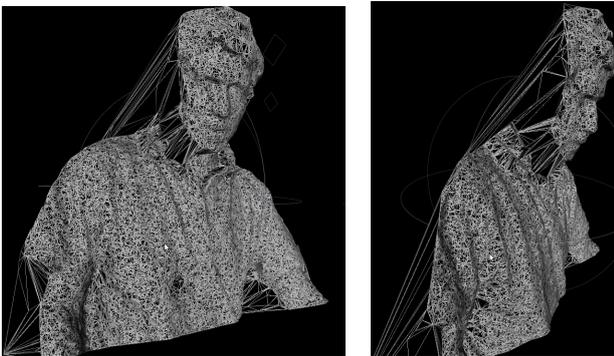


Figure 2. Mesh model using Crust triangulation

An incremental adaptation of Crust algorithm is proposed and enables the addition of new 3D points without having to recompute previous generated meshes. The stitching process relies on integrating new mesh poles as new vertices, on triangulation step and compute triangles only where both surfaces share vertices.

Given a set of registered points $X \in R^3$ sampled from an object surface S , it is possible to approximate its shape by a triangle mesh. The approach, based on a modified Crust algorithm [Amenta 98], uses a set of points P from the medial axis (poles) to extract a subset from the Delaunay triangulation of X that approximate S . The pole points, obtained from the Voronoi vertex or triangles average outer normal's, are positive (p^+) if they lie on the convex side of the surface and negative (p^-) otherwise. Once computed the Delaunay triangulation of $X \cup P$, the surface mesh is

estimated by extracting the set of simplices whose vertices belong to X . The proposed approach adds an incremental characteristic to the Crust algorithm as it is efficient viable to add new vertices to a Delaunay triangulation.

Assuming that a set of points X_t were already processed by the Crust algorithm, the set of poles P_t and the Delaunay triangulation are also available [Almeida 11]. To add a new set of sample points X_{t+1} to the surface mesh, avoiding a complete mesh recalculation, the following steps are performed:

Algorithm 1 Crust incremental algorithm

- 1: P_{t+1} =poles of X_{t+1}
 - 2: Add $P_{t+1} \cup X_{t+1}$ as new Delaunay triangulation vertices
 - 3: Extract triangles whose vertices belong to $X_t \cup X_{t+1}$
-

The procedure can be applied repeatedly to accommodate any number of point sets X_i . Nevertheless to avoid progressive grow in the number of mesh vertices, points closest to the mesh vertex (i.e. under a given Euclidean distance threshold) are deleted from the input point cloud before the incremental Crust step. Figure 2 illustrates a mesh model using the Crust approach.

Multiview 3D Scan: Recent depth sensor devices, like XBOX Kinect provide 3D measurements and also RGB data, enabling the use of 2D image algorithms. It is possible to improve the 2D feature mapping between consecutive RGB images, associating the respective depth data and creating a 3D feature tracking. The Xbox 360® Kinect™ Sensor combines a RGB camera and a structured light 3D scanner, consisting of an infrared camera and an infrared (IR) laser projector. The depth measurement principle is based on a triangulation process [Freedman 10].

Registration:

The registration process enables to align several 3D point clouds into one same referential to create a global model (Figure 3(b)). To register new 3D point clouds, acquired from different point of views, we perform algorithm 2 steps (Figure 3(a)):

Algorithm 2 Registration algorithm

- 1: Select one 3D point cloud shape to be the approximate 3D mean shape (ex: scan 0).
 - 2: Align the 3D point cloud shapes:
 - Compute the centroid of each 3D point cloud shape (or set of invariant features).
 - Align all shapes centroid to the origin.
 - Normalize each shapes centroid size.
 - Compute the rigid-body transformation using expression (5) to (7) to obtain the rotation R and translation t which best aligns both 3D shapes.
 - 3: Apply the calculated transformation to obtain new approximate 3D mean shape
-

Considerer the existence of two corresponding 3D points

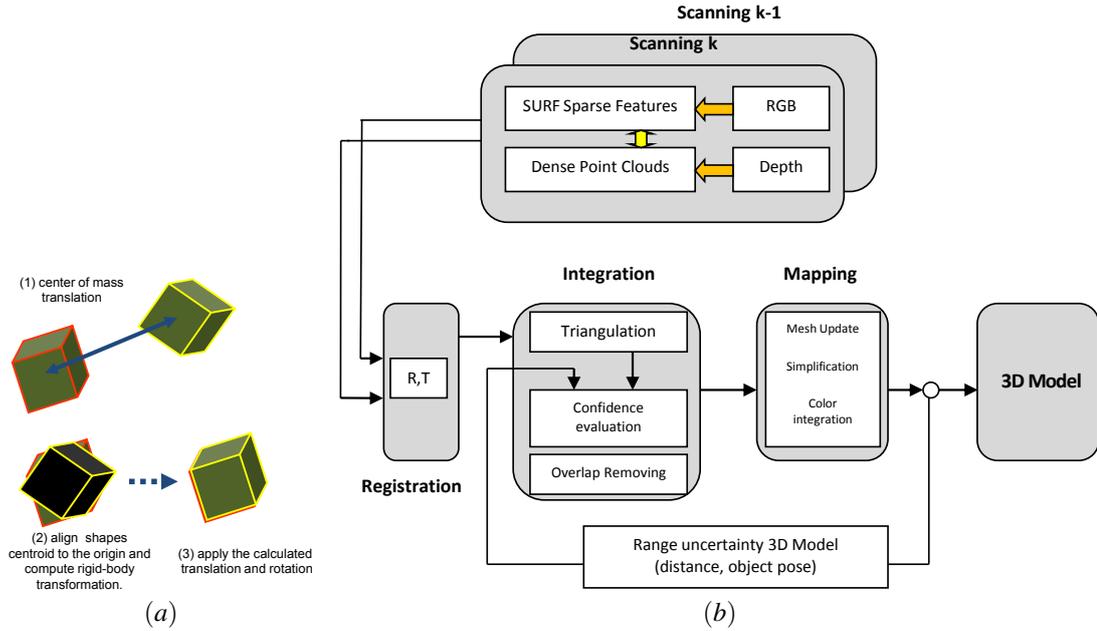


Figure 3. (a) Registration simplified flow. (b) Algorithm overview modules

sets $\{\mathbf{x}_i^t\}$ and $\{\mathbf{x}_i^{t+1}\}$, $i = 1..N$, from consecutive t and $t + 1$ scans, which relationship is given by equation (1):

$$\mathbf{x}_i^{t+1} = \mathbf{R}\mathbf{x}_i^t + \mathbf{t} + \mathbf{v}_i \quad \varepsilon^2 = \sum_{i=1}^N \left\| \mathbf{x}_i^{t+1} - \mathbf{R}\mathbf{x}_i^t - \mathbf{t} \right\|^2 \quad (1) \quad (2)$$

\mathbf{R} represents a standard 3x3 rotation matrix, \mathbf{t} stands for a 3D translation vector, and \mathbf{v}_i is a noise vector. The optimal transformation \mathbf{R} and \mathbf{t} that maps the set $\{\mathbf{x}_i^t\}$ on to $\{\mathbf{x}_i^{t+1}\}$ can be obtained through the minimization of the equation (2) using a least square criterion. The least square solution is the optimal transformation only if a correct correspondence between 3D point sets is guaranteed. Complementary methods are used to robust the correspondence (e.g. RANSAC). The singular value decomposition (SVD) of a matrix can be used to minimize Eq. (2) and obtain the rotation (standard orthonormal 3x3 matrix) and the translation (3D vector) [Arun 87][Challis 95][Eggert 97]. In order to calculate rotation first, the least square solution requires that $\{\mathbf{x}_i^t\}$ and $\{\mathbf{x}_i^{t+1}\}$ point sets share a common centroid. With this constraint a new of equation can be written using the following definitions:

$$\bar{\mathbf{x}}_i^t = \frac{1}{N} \sum_{i=0}^N \mathbf{x}_i^t \quad \bar{\mathbf{x}}_i^{t+1} = \frac{1}{N} \sum_{i=0}^N \mathbf{x}_i^{t+1} \quad (3)$$

$$\mathbf{x}_{ci}^t = \mathbf{x}_i^t - \bar{\mathbf{x}}_i^t \quad \mathbf{x}_{ci}^{t+1} = \mathbf{x}_i^{t+1} - \bar{\mathbf{x}}_i^{t+1} \quad (4)$$

$$\varepsilon^2 = \sum_{i=1}^N \left\| \mathbf{x}_{ci}^{t+1} - \mathbf{R}\mathbf{x}_{ci}^t \right\|^2 \quad (5)$$

Maximizing $Trace(\mathbf{R}\mathbf{H})$ enable us to minimize the generated equation (5), with \mathbf{H} being a 3x3 correlation matrix

defined by $\mathbf{H} = \mathbf{x}_{ci}^{t+1}(\mathbf{x}_{ci}^t)^T$. Considering that the singular value decomposition of \mathbf{H} results on $\mathbf{H}=\mathbf{U}\mathbf{D}\mathbf{V}^T$, then the optimal rotation matrix, \mathbf{R} , that maximizes the referred trace is $\mathbf{R}=\mathbf{U} \text{diag}(1; 1; \det(\mathbf{U}\mathbf{V}^T)) \mathbf{V}^T$:

$$\mathbf{R} = \mathbf{U}\mathbf{V}^T \quad (6)$$

The best translation that aligns $\{\mathbf{x}_i^{t+1}\}$ centroid with the rotated $\{\mathbf{x}_i^t\}$ centroid is

$$\mathbf{t} = \bar{\mathbf{x}}_i^{t+1} - \mathbf{R}\bar{\mathbf{x}}_i^t \quad (7)$$

Model Mapping

Suppose that the mapping from the world coordinates to one of the scans of the sequence, is known (ex: scan 0) and it is represented by the transformation ${}^0\mathbf{T}_w$. As described before, for any consecutive pair of scans (t , $t+1$) from tracked points it is possible to estimate rotation and translation and combine them into a single homogeneous matrix 4x4, ${}^{t+1}\mathbf{T}_t$, $\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}$.

Therefore it is possible to compute equation (8):

$${}^i\mathbf{T}_0 = {}^i\mathbf{T}_{i-1} {}^{i-1}\mathbf{T}_{i-2} \dots {}^1\mathbf{T}_0 \quad {}^i\mathbf{T}_w = {}^i\mathbf{T}_0 {}^0\mathbf{T}_w \quad (8)$$

To update the reconstructed model, each acquired 3D point set is transformed to the world coordinate system using ${}^i\mathbf{T}_w$. This alignment step adds a new scan to the dense 3D model. Alignment between successive frames enables to track the body position over small displacements.

Correspondence: the described 3D registration method requires the knowledge of point correspondences between the existing 3D points set and the newly acquired point

set. To solve this correspondence problem, we take advantage of the fact that RGB-D sensor provides simultaneously scene 3D information and respective 2D image. We propose the use of "Robust Image Features" (like Bay's Speeded Up Robust Features (SURF) [Bay 06]), which enables the identification of one same point in consecutive images. The association of a visual feature with its 3D point, enables to establish a match between consecutive 3D point clouds.

Although the SURF features enable the establishment of correspondences between points from both sets, illumination and viewpoints changes, together with sensor noise, among others, induce variations on those extracted features that may contribute to errors in the pairing process. This may indeed destroy the transformation estimation process by introducing unacceptable error or leading to no solutions.

For this reason we use the RANSAC algorithm [Fischler 81] to remove false correspondent point pairs that wrongly biases the rigid body transformation estimation. The approach randomly samples three 3D points correspondent pairs from consecutive scans and iteratively estimates the rigid body transformation [Arun 87] until find enough consensus or reach a maximum number of iteration based on the probability of outliers.

The registration method with outliers removal is described in following algorithm 3.

Integration: A new 3D mesh acquired from a different point of view and registered into a 3D global model can lead to two situations: (1) some *non-overlapped* triangles contains new information for the 3D model and (2) some *overlapped triangles* might contain redundant data, or more confident data useful for the model refining. To choose which information is relevant, we evaluate the data based on the uncertainty of range sensor. Sensor accuracy measures are dependent on the incident angle between the measuring ray and the surface distance.

Overlapping segmentation, front face checking and matching: the overlapping region is determined by projecting the pre-built mesh vertices's into the sensor 2D plane, once transformed for the referential of the newly scanned vertices and by checked the intersection area. We could simply re-triangulate all the points on the overlapping region, but due misalignment errors it can result on a bumpy surface. To tackle this challenge we propose an approach, where the triangulations update only happens if it contributes to improve the global model. The process consist in detecting overlapping triangles on the previous scanned range data image and the newly scanned range, and then keep those that provide more information for the model. We associate to each triangle a confidence value based on the measure uncertainty of its 3D vertices. The distance from where sensor acquires the data and the angle from it stands in front a surface are inversely proportional to the confidence (eq. 9):

Algorithm 3 Registration algorithm with outliers removal

```

1: Input : $X_p, X_q$ 
   {assumed correspondent 3D point pairs}
2: Output : $[R, t]$ 
   {rigid body transformation estimation}
3: while ( $i < MAXITER$ ) do
4:   randomly select 3 pairs of points
5:    $[R_i, t_i] \leftarrow$  estimate 6DOF rigid body transformation
   for these 3 pairs
6:    $X'_q = R_i * X_q + t_i$ 
   {apply the transformation to  $X_q$  scan to map it into
    $X_p$  reference frame}
7:    $inliers_i = |(X'_q - X_p) < \tau|, number\_of\_inliers_i$ 
   {determine the set of data points which are within a
   Euclidean distance threshold  $\tau$ }
8:   if ( $sizeof(inliers_i) > T_{threshold}$ ) then
9:      $[R, t] \leftarrow$  re-estimate the transformation model using
   all  $inliers_i$ 
10:    EXIT
11:   end if
12:   if ( $number\_of\_inliers_i > bestscore$ ) then
13:      $bestscore \leftarrow number\_of\_inliers_i$ 
14:      $best\_inliers \leftarrow inliers_i$ 
   {store cardinality of  $inliers_i$  and  $inliers_i$ }
15:     update  $MAXITER$ 
16:   end if
17:    $i = i + 1$ 
18: end while
19:  $[R, t] \leftarrow$  re-estimate the transformation model using all
   points from  $best\_inliers$ 

```

$$C_i = \left| \frac{1}{L\theta} \right| \quad (9)$$

where L is the distance between a 3D point and the range sensor's optical center and θ represents the sensor's pose angle in relation to the surface.

The angle θ is given by equation (10)

$$\theta = \arccos(\vec{n}_i, \vec{r}_i) \quad (10)$$

where \vec{n}_i is the normal of a triangle and \vec{r}_i is the normalized measurement ray from the sensor's optical center to the point.

The confidence measures capture the fact that points close to the sensor, as surfaces close to a fronto-parallel orientation, are typically captured more accurately by range sensors. The normal vector of a point consists of averaging normal vector of triangles formed with pairs of neighbors, and for each new scanned 3D mesh, a list of triangles (3D faces) is tagged with confidence information related with its 3D point positions. Integration of new triangles will occur, only if, its confidence contributes to improve the 3D model.

Figure 4 depicts the principle of a range sensor, composed by 3 ray measure beams, scanning an object from different positions (2D example). In this case, the range sensor acquires data from 4 different point of views, S_0, S_1, S_2, S_3 . For example, due overlapping data measures, between S_0, S_1, S_3 we can incrementally update the global model with the more confident edges (ex: P_{30}, P_{31}, P_{32}).

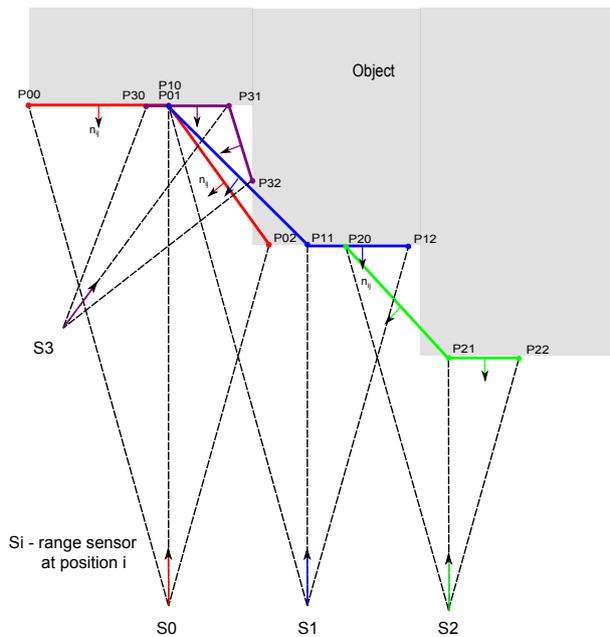


Figure 4. Range sensor, composed by 3 ray measure beams, scans an object from different positions (2D example)

Filtering Methods: depth maps containing holes, inconsistent data in the depth image object boundaries and vibrating behavior at the depth pixel level should be addressed to improve 3D reconstructions. Temporal filtering methods based on time data averaging clearly improves the depth maps quality, although are impractical on real-time applications or where moving objects exist. Several noise removal methods are possible to enhance the Kinect depth maps quality [Tomasi 98][Paris 06], like median filter, bilateral filter, joint bilateral filter, non-local means filter or moving square fitting. For example, the bilateral filter is a non-linear filter based on Gaussian distribution, which reduces the noise smoothing the signal while preserving the edges, however it has a high computational cost.

3 RESULTS

The integration and mesh refining algorithm were previously tested in matlab with noise free point data set and provided useful hints to understand the system. Figure 5 depicts a 3D mesh model of an object (light blue) for which the face triangles normals were computed (red arrows). These triangles and vertices's are projected into the RGB-D sensor plane, here represented by the light green square. The coordinate referential is composed by the blue axes and its intersection is the projection center (referential origin). The face triangles projections are represented in yellow. In Figure 6, the object is rotated slightly around its axes, here represented by light green color. Knowing the rigid transformation, the visible vertices are transformed to match with the previous model and reprojected into the sensor plane, Figure 7. The re-projection of the mesh into image sensor plane enables to detect the triangle intersection and preserve triangles with higher confidence.

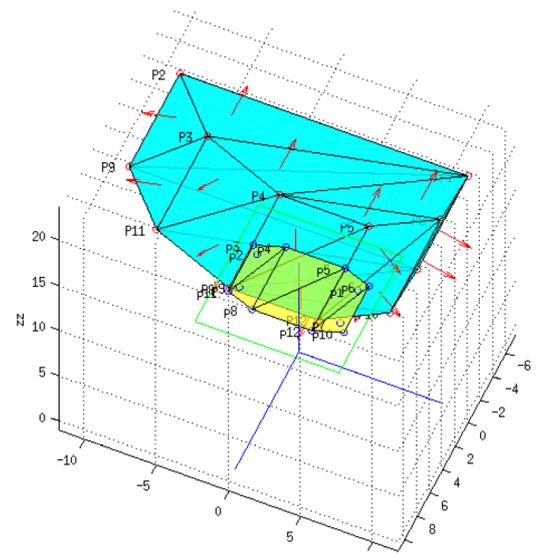


Figure 5. Fixed range sensor scanning an object

In Figure 9 we show an example of correspondence between consecutive image features using SURF method (white lines indicate correspondent point).

Figure 10 depicts a sequence of scans that creates a 3D per-

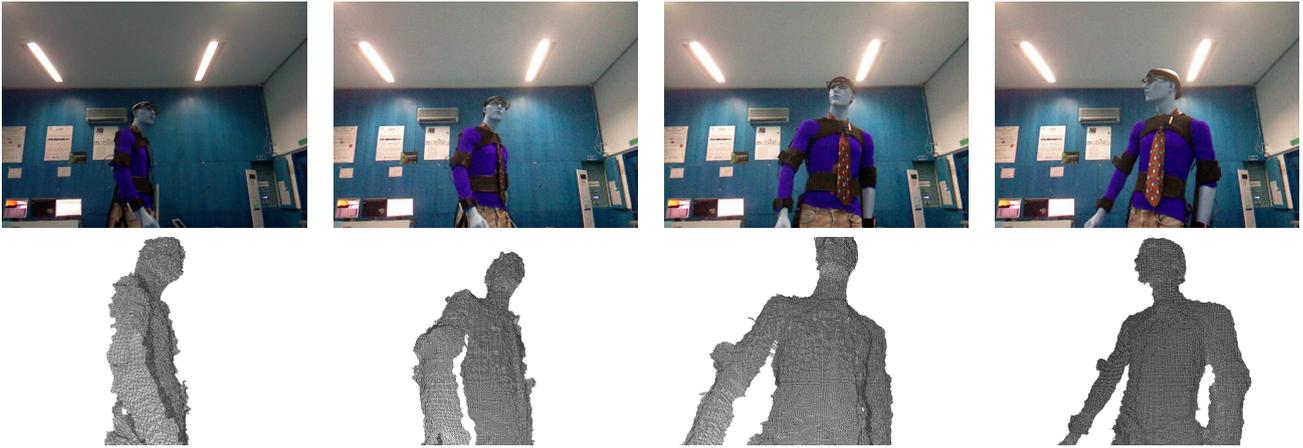


Figure 10. Sequence of mesh models to be integrated, triangulation based on depth data sensor grid structure and depth information.

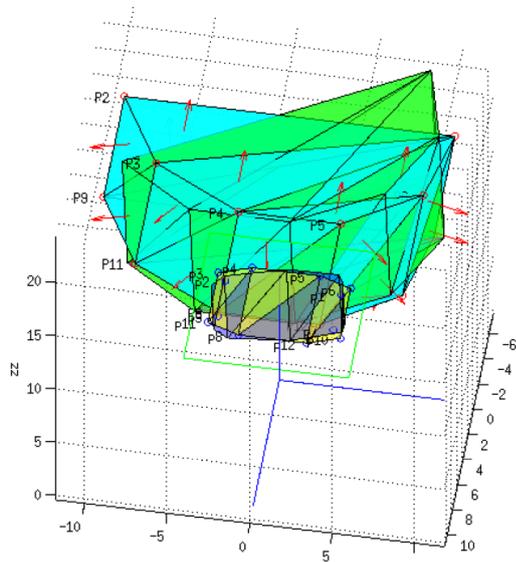


Figure 6. Moving object

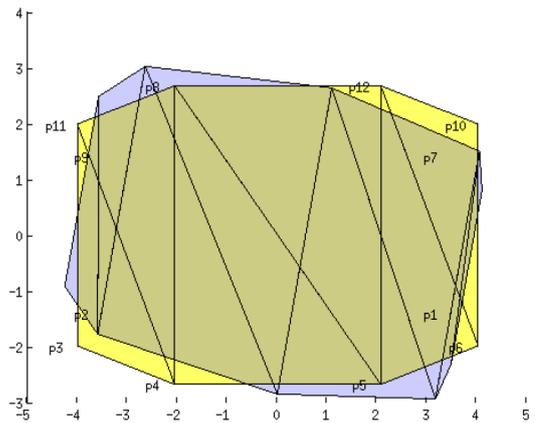


Figure 7. Mesh re-projection into image sensor plane to detect triangle intersection. Preserve triangles with higher confidence.

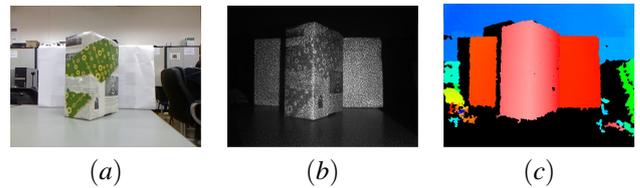


Figure 8. (a) RGB image. (b) IR monochromatic image with speckles pattern projected onto a scene. (c) Depth map with distances associated to colors.

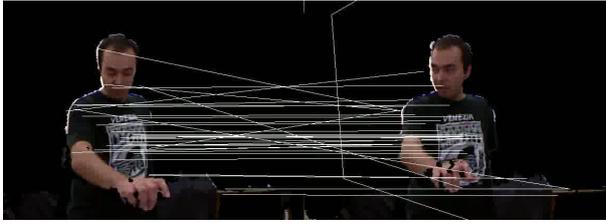


Figure 9. SURF features matched on consecutive time frames

son model. On the top row we present RGB images of the scene and in lower row snapshots of the respective meshes, generated in real time. The mesh triangulation is based on depth data sensor grid structure and depth information. To achieve the real time characteristic, we programmed in OpenGL for Embedded Systems (OpenGL ES) as it enables vertex buffers to be processed in parallel as a single entity. GPU shaders and OpenCV [OpenCV 15] were also used.

Figure 11 shows a reconstructed 3D model. It results from several 3D point clouds fused in real time after applying successive 3D rigid body transformations, mesh refining integration and rendering.

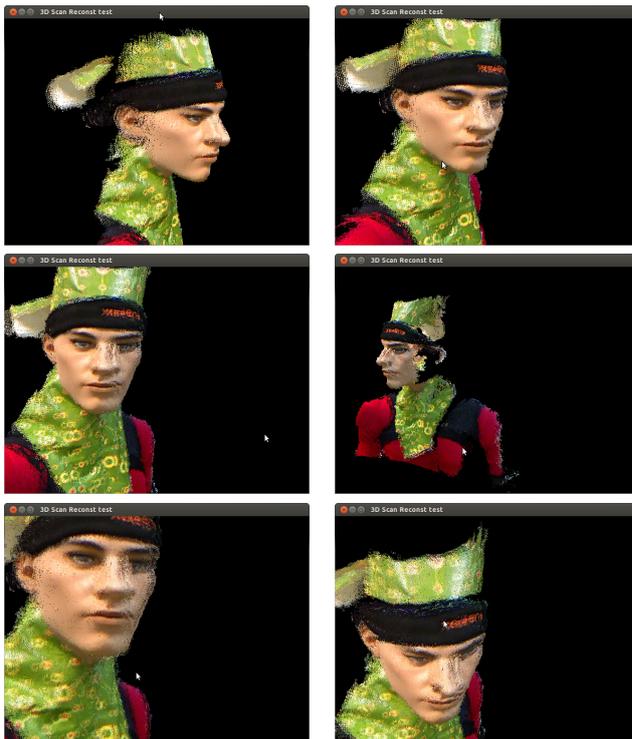


Figure 11. Synthesized views of a on-line 3D reconstructed model dependent of observer point of view.

3.1 Discussion

Processing real data allowed us to identify some noise sources that can affect the algorithm. For example, SURF

points can generate erroneous matches due image noise and they are more common on body boundaries (Figure 9 presents some wrong diagonal links for an almost pure vertical axis body rotation). The body to be reconstructed should be segmented from background static areas using a motion filter. Scale-invariant feature transform (SIFT) [Lowe 04] was also tested and presented better accuracy as key feature descriptor, although we have chosen SURF method in order to achieve the real-time characteristic. The kinect system imaging geometry introduces structural errors that are function of the distance to the object and the sensor orientations relative to the object surface. A proper calibration of the RGB-D sensor is essential to improve results. Stereo calibration procedures were used to estimate the intrinsic parameters of both RGB and IR (depth) cameras, as the relative transformation (R, T) between them. The estimated camera's parameters and transformations enabled us to align Kinect™ both RGB with IR (depth) cameras and obtain more reliable data information (as depicted in Figure 8). The proposed reconstructed 3D model approach enables to generate any virtual synthesized view for an observer that moves in front of a display, that is, a required augmented reality (AR) functionality.

4 CONCLUSION

A free viewpoint system framework is proposed to generate view dependent synthesis based on scene 3D mesh model. Our approach explores virtual view synthesis through motion body estimation and hybrid sensors composed by video cameras and a low cost depth camera based on structured-light. The solution addresses the geometry reconstruction challenge from traditional video cameras array, that is, the lack of accuracy in low-texture or repeated pattern region. We present a full 3D body reconstruction system that combines visual features and shape-based alignment. Modeling is based on meshes computed from dense depth maps in order lower the data to be processed and create a 3D mesh representation that is independent of view-point. Research contributions include a new incremental version of Crust algorithm that efficiently adds new vertices to an already existing surface without having to recompute previous generated meshes and a topological incremental reconstruction approach based on confidence measures that avoids redundant data information computation.

With this on-line reconstructed 3D model, we can provide synchronous point of view for an observer that moves in front of a display of a face-to-face meeting application, thus enhancing the presence sensation. Future work includes framework usability tests for a telepresence meeting application. This work presents an on-line incremental 3D reconstruction framework that can be used on low cost telepresence applications, augmented reality (AR) or human robot interaction applications.

References

- [Almeida 11] Luis Almeida, Filipe Vasconcelos, João Barreto, Paulo Menezes, and Jorge Dias.

- On-line incremental 3d human body reconstruction for hmi or ar applications. In *CLAWAR 2011: 14th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machine*. Paris, France, September 2011.
- [Almeida 13] Luis Almeida, Paulo Menezes, and Jorge Dias. *Handbook of Research on ICTs for Human-Centered Healthcare and Social Care Services*, chapter Augmented Reality Framework for the Socialization between Elderly People, pages 430–448. IGI Global, 2013.
- [Amenta 98] Nina Amenta, Marshall Bern, and Manolis Kamvyselis. A new voronoi-based surface reconstruction algorithm. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, pages 415–421. ACM, New York, NY, USA, 1998.
- [Arun 87] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9:698–700, September 1987.
- [Bay 06] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *In ECCV*, pages 404–417. 2006.
- [Beck 13] S. Beck, A. Kunert, A. Kulik, and B. Froehlich. Immersive group-to-group telepresence. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):616–625, 2013.
- [Challis 95] J. Challis. A procedure for determining rigid body transformation parameters. *Journal of Biomechanics*, 28(6):733–737, jun 1995.
- [Eggert 97] D. W. Eggert, A. Lorusso, and R. B. Fisher. Estimating 3D rigid body transformations: a comparison of four major algorithms. *MACHINE VISION AND APPLICATIONS*, 9:272–290, 1997.
- [Fischler 81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395, June 1981.
- [Freedman 10] Barak Freedman, Alexander Shpunt, Meir Machline, and Yoel Arieli. Depth mapping using projected patterns, May 2010.
- [Henry 12] Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. *I. J. Robotic Res.*, 31(5):647–663, 2012.
- [Kurillo 08] G. Kurillo, R. Vasudevan, E. Lobaton, and R. Bajcsy. A framework for collaborative real-time 3d teleimmersion in a geographically distributed environment. In *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on*, pages 111–118. dec. 2008.
- [Lowe 04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60:91–110, November 2004.
- [Newcombe 11] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '11, pages 127–136, Washington, DC, USA, 2011. IEEE Computer Society.
- [OpenCV 15] OpenCV. <http://opencv.org>, 2015.
- [Paris 06] Sylvain Paris and Frédo Durand. A fast approximation of the bilateral filter using a signal processing approach. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part IV*, ECCV'06, pages 568–580, Berlin, Heidelberg, 2006. Springer-Verlag.
- [Petit 09] Benjamin Petit, Jean-Denis Lesage, Clément Menier, Jérémie Allard, Jean-Sébastien Franco, Bruno Raffin, Edmond Boyer, and François Faure. Multicamera real-time 3d modeling for telepresence and remote collaboration. *INTERNATIONAL JOURNAL OF DIGITAL MULTIMEDIA BROADCASTING*, 2010:247108–12, 2009.
- [Tomasi 98] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*, pages 839–846, Jan 1998.

Modelo de descrição de experiências multissensoriais multiutilizador

Rui Rodrigues

INESC TEC e DEI/Faculdade de Engenharia da Universidade do Porto
{rui.rodrigues,eduardom}@fe.up.pt, hugomachado@gmail.com

Eduardo Magalhães

Hugo da Silva

Maximino Bessa
INESC TEC e UTAD
maxbessa@utad.pt

António Coelho
INESC TEC e DEI/Faculdade de Engenharia da Universidade do Porto
{acoelho,aas}@fe.up.pt

Augusto Sousa

Resumo

Neste momento os ambientes virtuais estão a tornar-se numa peça importante nas aplicações centradas no utilizador onde se inclui o treino, design, entretenimento etc.

Apesar de percecionarmos o nosso mundo através dos nossos múltiplos sentidos, estas novas aplicações recorrem normalmente principalmente ao estímulo da visão, sendo o som o segundo estímulo mais explorado.

Descobertas recentes sugerem que quantos mais sentidos são estimulados num ambiente virtual mais imersivo é a experiência e o desempenho dos utilizadores aumenta (tal como acontece no mundo real).

O projeto MASSIVE pretende investigar em detalhe como é que podemos obter um ambiente virtual credível no qual os vários sentidos do utilizador são estimulados de forma a alcançar uma elevada imersividade.

Nesse contexto, foi desenvolvido um modelo para a descrição de experiências multissensoriais, a ser usado para representar não só as cenas, como as interações e estímulos do utilizador de forma a suportar contextos de interação multimodal e multissensorial, de forma modular. Esse modelo é apresentado neste artigo, juntamente com um exemplo prático de aplicação, demonstrando a sua relevância.

Keywords

Realidade Virtual, Experiências Multissensoriais, Modelo de descrição de dados

1. INTRODUÇÃO

O conceito de “presença” é um dos mais relevantes no contexto da realidade virtual. A sensação de presença é muito dependente do estímulo de vários sentidos, tal como acontece na vida real. Embora a evolução dos computadores, placas gráficas e o aumento da oferta de HMDs (*Head Mounted Displays*) a preços acessíveis e com qualidade aceitável tenham permitido aumentar os níveis de imersão, estes níveis continuam longe do necessário para convencer o utilizador de que realmente está “presente” num ambiente virtual. É sabido que a integração de estímulos de múltiplos sentidos, além dos mais comumente explorados - a visão e a audição - permite um salto qualitativo na sensação de “presença”.

Este tipo de ambientes virtuais multissensoriais tem sido por isso alvo de crescente interesse a nível de investigação. A sua aplicação em diferentes contextos, como o entretenimento, o treino e a certificação, o turismo e o mercado de experiências torna-os também atrativos comercialmente. No entanto, a sua implementação prática é ainda muito limitada e pouco acessível, dados os recursos e dispositivos envolvidos, e a dificuldade em criar conte-

údos para os mesmos que contemplem, por exemplo estímulos hápticos e olfativos.

Se a exploração destes ambientes por um único utilizador por si só é bastante atrativa, a possibilidade de integrar vários utilizadores em simultâneo apresenta-se como mais interessante ainda. A colaboração/competição suportada pelas redes sociais tornou-se numa mecânica comum em grande parte dos jogos. Este conceito social multi-utilizador vem sendo aplicado em ambientes virtuais como o *Second Life*¹ ou o *There*² e mantém - ou até aumenta - a sua relevância no que diz respeito a ambientes virtuais multissensoriais. O lançamento do *Oculus Rift DK2 Project Viewer*³ pela *Linden Lab* confirma o interesse das empresas responsáveis por ambientes virtuais multiutilizador na imersão proporcionada pela realidade virtual.

¹ <http://secondlife.com/>

² <http://www.there.com/>

³ <https://community.secondlife.com/t5/Featured-News/Oculus-Rift-DK2-Project-Viewer-Now-Available/ba-p/2843450>

Deste interesse na criação de ambientes virtuais multissensoriais multiutilizador, surge não só a necessidade de sistemas que implementem essas experiências, integrando os vários dispositivos de entrada e saída responsáveis pela interação multimodal e os estímulos multissensoriais associados, mas também a de mecanismos para descrever, criar, e armazenar este tipo de ambientes. São necessárias estruturas e modelos de dados, assim como ferramentas de autoria dedicadas a este tipo de experiências.

Estas lacunas, ainda por preencher, contribuem para a relevância da criação de modelos para cenários multissensoriais, como identificado por diversos autores [1,2,3,4] (ver secção 2). Em particular, é importante que um tal modelo permita definir características e elementos de uma experiência deste género a um alto nível, mas que tenha a flexibilidade para suportar um ou mais mapeamentos de diversos dispositivos, num contexto multiutilizador.

É nesse sentido que apresentamos neste artigo um modelo de representação de experiências para suporte ao desenvolvimento rápido de ambientes virtuais multissensoriais, independente do *hardware*. Um modelo deste tipo poderá permitir a geração de cenários virtuais multissensoriais e multiutilizador de uma forma relativamente expedita e disponível a utilizadores, permitindo por um lado construir bases de dados de cenários com iterações permanentes (adequando os dispositivos de controle e os conteúdos) e por outro, testar a robustez do sistema na execução desta tarefa por utilizadores “comuns”. Em conjugação com a ferramenta de autoria pode permitir a ligação direta entre a arquitetura física (motor multissensorial) e a conceptualização de experiências virtuais através da observação em tempo quase real dos objectivos pretendidos.

Esta abordagem partilha alguns dos conceitos utilizados nas arquiteturas de motor de jogo baseadas em componentes e *scripting*, como por exemplo o Kismet ou Blue Print do Unreal Engine, mas vai mais longe ao prever à partida componentes multissensoriais abstraídos, e a sua interoperabilidade com os sensores/dispositivos, a cena e os utilizadores.

Para testar a proposta, uma das provas de conceito do projeto MASSIVE é implementada segundo o modelo proposto. A experiência consiste numa descida de bicicleta por uma montanha, na qual o utilizador tem a percepção visual do ambiente campestre que o rodeia, ouvindo também os sons da paisagem sonora local, assim como pássaros ou os ruídos localizados da própria bicicleta. O aroma da montanha é também fornecido ao utilizador, assim como a trepidação do veículo ao deslocar-se sobre o solo e aquando de eventuais saltos. Em termos hápticos, é possível também sentir no peito o impacto com árvores e arbustos e o próprio vento natural da paisagem.

Este documento está organizado da seguinte forma: O trabalho relacionado é discutido na secção 2. A secção 3 descreve o conceito, requisitos e arquitetura do modelo de descrição de experiência proposto. A secção 4 está dedicada a uma proposta de descrição e armazenamento

de dados em formato XML para experiências multissensoriais e a secção 5 apresenta uma prova de conceito que consiste numa experiência de realidade virtual. A secção 6 conclui o artigo e inclui o trabalho futuro que se pretende realizar.

2. TRABALHO RELACIONADO

A necessidade de representar formalmente experiências multissensoriais tem vindo a ser discutida por diversos autores, apontando linhas de investigação e modelos semânticos diversos [2,3,4]. A vantagem destes modelos consiste na possibilidade de descrever relações multimodais entre objetos e entre utilizadores de uma forma adaptativa, proporcionando suporte a ferramentas de autoria versáteis e modulares. Desta forma é também possível uma abstração dos dispositivos que irão gerar os estímulos. Isto por um lado possibilita representar a mesma experiência utilizando conjuntos de dispositivos diferentes, mas por outro lado implica a adaptação automática do conteúdo, tendo em conta informação sobre os elementos multissensoriais de um cenário virtual, como por exemplo o número de fontes sonoras ou o tipo de *output* visual, e a inferência do seu mapeamento aos dispositivos necessários.

Gutierrez et. al [4] propõem um modelo adaptativo para Interfaces Multimodais (IMM), suportado na importância da adaptabilidade defendida por Xiao [5], que consiste em subdividir e sistematizar os cenários virtuais em duas entidades: dispositivos de interação (interfaces multimodais) e entidades virtuais (componentes 3D, imagens, texto, etc.). O modelo semântico proposto por estes autores é baseado em descritores que definem a interação entre dispositivos e mundos virtuais que, através de um sistema de canais de I/O, possibilitam que os objetos virtuais funcionem como sistemas reativos. Estes objetos poderão ter representações diversas tais como imagens, formas 2D e 3D, texto, vídeo, etc., prevendo funcionalidades “personalizáveis” ou escaláveis a novos contextos. A formalização do modelo está relacionada com a necessidade de interpretação do modelo tanto por humanos como por máquinas, pelo que propõe uma representação baseada em XML como suporte. Apesar de ser um modelo flexível, está focado essencialmente nos dispositivos de entrada e no seu mapeamento aos objetos, e não contempla de forma explícita a vertente da geração de estímulos multissensoriais

Num nível de abstração mais elevado, Irawati et al [2] descrevem uma *framework* baseada numa ontologia dos objetos que estabelece limites e regras de interação e relacionamento entre os objetos de um cenário virtual. Um dos principais objetivos é o de resolver ambiguidades relacionadas com a intermodalidade sensorial durante a manipulação de objetos 3D. Para tal propõem um modelo de descrição de cena que inclui não só as regras de interdependência entre objetos mas também as características espaciais destes mesmos objetos em relação a outros elementos que compõem o espaço. Um exemplo deste tipo de ambiguidades seria o utilizador, num dado cenário virtual, querer colocar um objecto “quadro” no objecto “parede” do cenário mas ter o objecto “homem” entre a

sua localização e o ponto para onde está a apontar. Se a relação entre o objecto “quadro” e o objecto “parede” não estiver descrita na ontologia, o sistema pode compreender que o objecto “quadro” é para ser colocado no objecto “homem” e não no objecto “parede”. Adicionalmente, o sistema permite também fazer integração de diferentes dispositivos de entrada, formando uma interface multimodal. No entanto, o mapeamento destes dispositivos não é flexível, não sendo contemplado na ontologia de objectos. Esta por sua vez contempla apenas informação espacial/tridimensional, que influencia apenas a componente visual, não incluindo outro tipo de estímulos.

Ho et al [3] descrevem uma ontologia baseada no RoSE (Representation of Sensory Effects) [9] que define as relações entre efeitos sensoriais e os dispositivos que produzem os estímulos, usando regras de inferência. Apresentam também uma aplicação - Ontology-based Multi-Sensory Inference Engine (OMSIE) - que tenta inferir estímulos a partir da informação dos efeitos sensoriais pretendidos presentes numa cena. Esta proposta é interessante no sentido de tentar automatizar a geração de estímulos através de regras. No entanto, não contempla a componente de interação multi-utilizador.

A própria proposta de standard MPEG-V inclui, na sua parte 3 [6], a proposta de uma linguagem baseada em XML para a descrição de efeitos sensoriais tais como luz, vento, nevoeiro, vibração, etc. - Sensory Effect Description Language (SEDL) [7]. Os efeitos propriamente ditos não fazem parte da SEDL, mas são definidos num vocabulário - Sensory Effect Vocabulary (SEV) - que permite abstrai-los de forma a serem instanciados em cada domínio específico, e poderem ser mapeados aos dispositivos propriamente ditos (ventoinhas, luzes, etc.). [8] apresenta um proposta de implementação, focada na extensão de sequências de vídeo/áudio com outros efeitos sensoriais. Tanto no MPEG-V como nesta implementação, o foco é na descrição dos efeitos, não contemplando propriamente a descrição de cenas e o mapeamento de efeitos a diferentes elementos de uma cena, ou a componente de interação.

Em suma, constata-se que apesar de haver por um lado, esforços no sentido de definir ontologias, modelos de dados ou formatos de descrição de cenas em ambientes interativos, e por outro de definir mecanismos de abstracção de efeitos sensoriais, há ainda muito em aberto no que diz respeito à descrição integrada de cenas interativas que contemplem estímulos multissensoriais, e multiutilizador.

3. MODELO DE DADOS

Na secção anterior é mencionada a importância da representação de relações entre dispositivos de entrada e mecanismos de interação, a descrição de uma cena contemplando estímulos e o seu mapeamento a dispositivos de saída e suas características, bem como a relação com vários utilizadores.

Este é o ponto de partida para o modelo de descrição de experiência proposto neste artigo que se pretende sirva como forma de descrição de experiências multissensoriais interativas, assim como de suporte a ferramentas de

autoria deste tipo de conteúdos. O *workflow* esperado é o seguinte: alguém que pretenda criar uma experiência multissensorial usa uma ferramenta de autoria que gera um ficheiro descrevendo a experiência segundo o modelo aqui proposto; esse ficheiro é carregado por um sistema multissensorial, ao qual estarão ligado(s) o(s) utilizador(es); o sistema irá interpretar as interações dos utilizadores e gerar os estímulos adequados, de acordo com a descrição da experiência e os dispositivos de estímulos que tiver disponíveis.

O modelo prevê uma arquitetura modular por cinco tipos de entidade para descrever experiências multissensoriais: *Experiment*, *Scene*, *User*, *Inputs* e *Renderers* (figura 1). Em particular, é explorado o padrão de arquitetura baseada em componentes, de forma a possibilitar a integração de diferentes fontes de estímulos a diferentes elementos da cena.

A experiência (*Experiment*) é o bloco nuclear que agrega os restantes componentes e pode ser constituída por várias cenas. Uma cena (*Scene*) será o equivalente a um espaço/tempo, sendo que, no contexto de uma mesma experiência, um utilizador pode passar por várias cenas, mas que num dado instante apenas poderá estar presente numa cena. Um utilizador (*User*) é simultaneamente:

- um ator na cena com uma representação a vários níveis (visual, sonora, etc.),
- controlado por um utilizador real através de diferentes canais/dispositivos de entrada (*Inputs*)
- passível de influenciar/interagir com outros elementos da cena
- um recetor dos estímulos multissensoriais resultantes da experiência, gerados através de geradores desses estímulos (*Renderers*) que em última instância controlarão dispositivos de saída (HMD's, altifalantes, ventoinhas, etc.).

Estará, portanto, geralmente representado por um objecto na cena de forma a permitir a sua visualização e localização no espaço (por exemplo por outros utilizadores).

Cada uma das entidades é descrita mais detalhadamente nas subsecções seguintes.

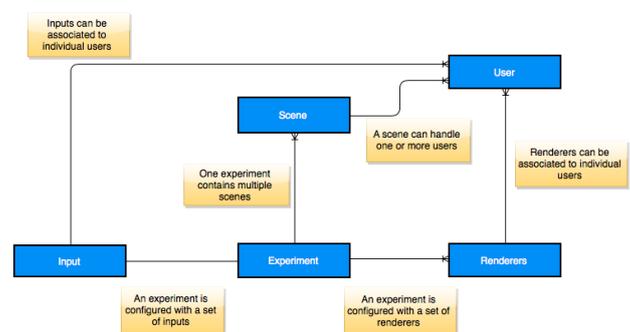


Figura 1 – Diagrama.

3.1 Scene

A cena recebe um conjunto de características de ambiente do qual constam propriedades como o local (que poderá ser baseado no mundo real ou num conceito puramente

sintético), momento temporal, condições ambientais ou os próprios valores que ditam a gravidade, entre outras propriedades relevantes.

Uma cena pode conter objetos e utilizadores, sendo que um utilizador pode estar associado a um objecto (figura 2). Árvores, veículos, animais ou mesmo humanos geridos por inteligência artificial são considerados objetos. Como tal, estes terão influência na experiência sendo visualizados, emitindo sons, cheiros ou afetando fisicamente outros objetos e utilizadores. Para tal, um objeto compreende sempre uma transformação que indica a sua posição, rotação e escala no espaço. Os objetos podem ser “pais” de outros objetos, permitindo assim situações como o exemplo em que um objeto “cesta” contenha um ou vários objetos “pão”, constituindo assim a este nível um típico grafo de cena.

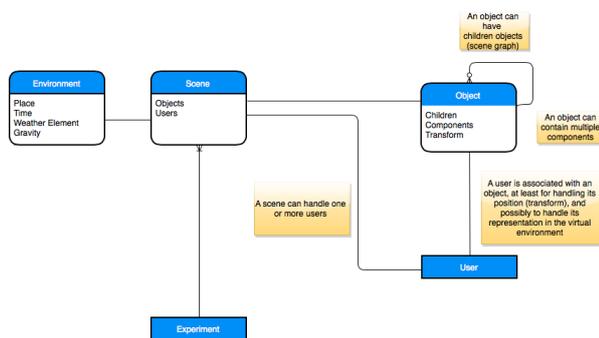


Figura 2 - Sistema relacional de Cena.

Um dos factores distintivos desta proposta é a sua arquitetura baseada em componentes. Qualquer objeto pode conter um conjunto de componentes de diferentes tipos, que definem características como as suas propriedades físicas, aroma associado, ou outras. Estas propriedades serão posteriormente utilizadas no cálculo dos impulsos fornecidos ao utilizador pelos renderizadores que lhe estejam aplicados. Com isto consegue-se uma abstração dos efeitos sensoriais, ao estarem associados aos elementos da cena, ao invés de serem globalmente declarados e despoletados. Por exemplo, uma chávena de café próxima de um utilizador dará origem à emissão do cheiro de café com uma intensidade inversamente proporcional à distância entre ambos.

O modelo contempla um conjunto de componentes que cobrem diferentes tipos de estímulos (ligados à componente visual, auditiva, háptica e olfativa). No entanto, além destes, novas ações e comportamentos podem ser adicionados por um componente de *scripting*, que poderá também permitir a recepção de dados de *input*, a interação com outros objetos, ou outros comportamentos mais específicos (figura 3).

3.2 Users

Como referido anteriormente, um utilizador no modelo proposto é um ator representado na cena, controlado por um utilizador real e que pode interagir com a cena, e receber estímulos da mesma, resultantes da experiência. Estará, portanto, geralmente ligado a um objeto de forma a permitir a sua visualização e localização no espaço - avatar. Múltiplos utilizadores podem coexistir dentro da

mesma experiência. Isto permite experiências colaborativas ou competitivas, em que, por exemplo, um utilizador poderá estar numa bancada a ver um jogo de futebol, enquanto outros estarão a experimentar as sensações de jogar o jogo no próprio campo.

Os estímulos que são fornecidos a um dado utilizador serão gerados em tempo real, tendo em conta as fontes e influenciadores virtuais de estímulos (p.ex. uma chávena de café, a relva do campo, o vento), a sua relação espacial com o utilizador em causa (distância, orientação, etc.), e os dispositivos de estímulos disponíveis (HMD, máquinas de cheiros, ventoinhas, etc.), entre outros fatores.

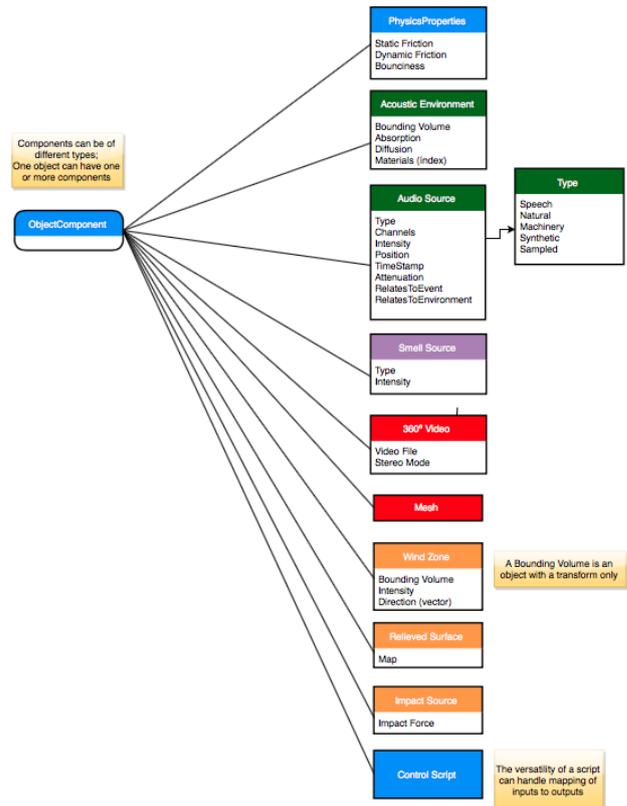


Figura 3 - Object Component e Script Component.

Esse mapeamento e geração são geridos pelos renderizadores que, numa mesma experiência, podem ser diferentes para diferentes utilizadores (p.ex., um utilizador associado a um jogador no campo pode usar um HMD para a componente visual e um colete para receber estímulos hápticos, enquanto outro utilizador associado a um espectador pode usar um monitor e não ter estímulos hápticos).

3.3 Inputs

As entradas (*inputs*) representam canais de input de médio/alto nível através dos quais um utilizador real pode controlar o seu avatar, mas que também podem ser usados para controlar outros elementos da cena (p.ex. sensor que determine a temperatura da sala real em que o utilizador se encontra). Por essa razão, as entradas podem ser diretamente associadas a utilizadores ou à própria experiência (e consequentemente a cenas individuais).

Estas entradas podem ser alimentadas por dispositivos HID (*Human Interface Device*), sensores de baixo nível,

deteção de áudio, dispositivos de *tracking* ou mesmo sensores de dados biométricos (Fig. 4). O mapeamento entre parâmetros de dispositivos de entrada e objetos ou a própria experiência simplifica o controlo, permitindo que de uma forma direta sejam conseguidos efeitos diretos (como por exemplo a translação de um utilizador aquando da pressão de uma tecla) ou indiretos, em que um sensor LDR (*Light Dependent Resistor*) permita por exemplo utilizar a luz ambiente do espaço real para controlar o ciclo dia-noite.

Mapeamentos mais diretos, como o *tracking* do utilizador real para a orientação do seu avatar podem ser representadas implicitamente mapeando o *Input* ao Utilizador. Outros mapeamentos podem ser representados recorrendo à utilização de um componente de *scripting* (descrito na secção 4) num objeto, permitindo que este seja controlado ou parametrizado por dispositivos de entrada de dados.

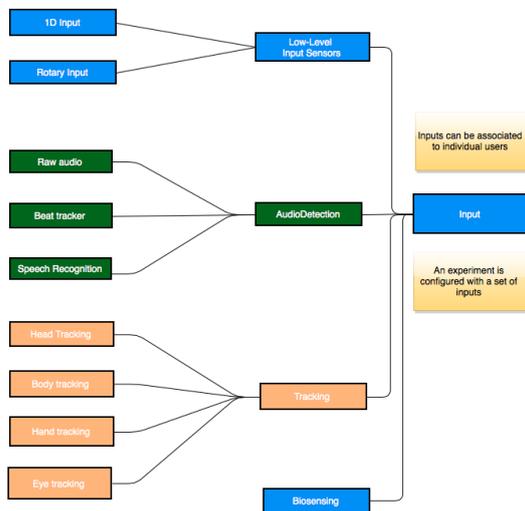


Figura 4 - Sistema de entradas através de dispositivos HID.

3.4 Renderers

Os renderizadores (*Renderers*) são as entidades responsáveis pela geração dos estímulos dependentes do utilizador, sendo portanto uma abstração dos dispositivos de saída/estímulos. Com base nos fatores ambientais, geradores de calor/frio, som, impacto e demais impulsos sensoriais, um renderizador é responsável por calcular o efeito destes fatores sobre o utilizador dependendo das condições do seu avatar e, de uma forma independente do hardware, providenciar os impulsos sensoriais ao utilizador. Uma classe de renderizadores foi identificada para cada um dos cinco sentidos e, para cada sentido, foram identificadas subclasses destes para diferentes tipos de dispositivos. Por exemplo, para a visão, pode ser associado um renderizador de uma única vista (“mono”, para visualização num monitor normal) a um dado utilizador, e um renderizador stereo (para um HMD 3D) a outro utilizador (Figura 5).

4. FORMATO DE FICHEIRO

Com base no modelo proposto, foi delineado um *schema* de XML, que permite descrever uma experiência com

base nos parâmetros estabelecidos. Este formato permite a utilização transparente entre diferentes sistemas e apresenta a vantagem de ser perceptível pela máquina, mas também legível por humanos.

Como raiz utilizamos o elemento <Experiment>, que representa toda a experiência e utiliza o atributo name para a identificar. Mais concretamente, tem como subelementos (descendentes diretos) os três elementos principais <renderers>, <inputs> e <scenes>, e um quarto elemento <controlScripts> para definir comportamentos adicionais via *scripting* (figura 7).

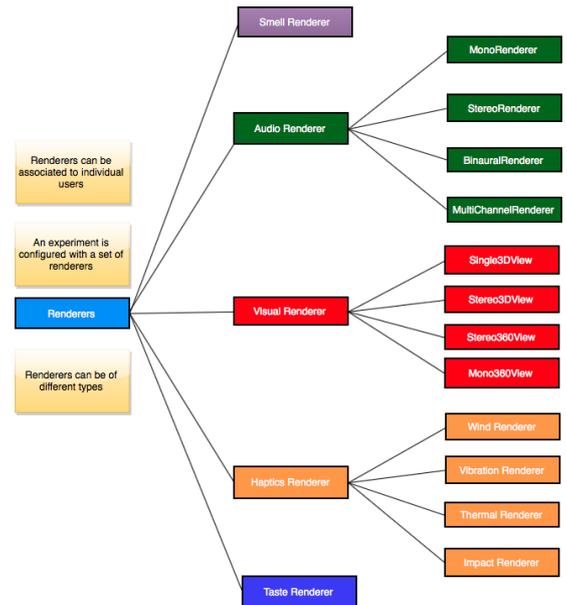


Figura 5 - Diferentes tipos de renderers.

No nó <renderers> são instanciados os renderizadores a utilizar na experiência (figura 6, secção A). Cada renderizador é definido num nó que o identifica, acompanhado de um identificador. Poderá existir também um atributo que define uma subclasse específica com informação suplementar, caso seja relevante para o renderizador.

Em <inputs> (figura 6, secção B) são descritos os dispositivos de entrada de dados utilizados. Uma vez mais, um atributo id identifica o dispositivo, e cada sub-nó poderá conter atributos específicos, caso tal seja necessário.

No elemento <Scenes> são descritas as várias cenas que compõe a experiência (figura 6, secção C). Cada cena contém utilizadores, objetos, ambientes, mas também referências aos renderizadores, inputs e demais elementos definidos das secções relevantes. A enumeração e identificação dos elementos necessários para a constituição de uma cena recorre a vários nós que descrevem o seu conteúdo.

Os utilizadores são descritos no elemento <Users>, com um subelemento <User> para cada um com o id que os identifica (figura 6, secção C.1). Caso um utilizador tenha um objeto associado, uma referência para esse objeto é indicada no atributo objectRef. Os renderizadores associados ao utilizador são referenciados na sua própria secção

<renderers>, com o elemento <RenderRef> a indicar no seu id o renderizador a utilizar.

Os *inputs* a que responde cada utilizador são da mesma forma indicados na sua secção <inputs>, sendo que cada um é referenciado pelo elemento <InputRef> que inclui a indicação do id referente à entrada relevante.

A secção *objects* (figura 6, secção C.2) contém a informação dos vários objetos presentes na cena. Cada objeto é identificado pelo elemento <object> que inclui como atributo o seu id. A posição, rotação e escala iniciais da sua transformação são definidas pelo elemento <Transform> com os atributos Position, Rotation e Scale utilizados para a indicação dos valores relevantes. A estrutura hierárquica é também fornecida no elemento object, com o(s) objeto(s) filho(s) a aparecer na secção <children>. Componentes podem também ser aqui referenciados com o elemento <ObjectComponent> que inclui a sua classe no atributo *class*. <ObjectComponent> pode também conter nós para definir scripts no nó <Script>. Subelementos do nó <ObjectComponent> podem também conter parâmetros incluídos nos atributos do elemento <param>. <MeshComponent> será o elemento utilizado sempre que um objeto tenha uma *mesh* associada.

Scripts referenciados em outras áreas do ficheiro XML podem ser definidos na secção ControlScripts (figura 6, secção D). Para isso, o elemento <ControlScript> é identificado por um id utilizado no atributo scriptRef do elemento <Script> que o utiliza. Parâmetros como valores por omissão podem ser definidos no subelemento <param>. O código pode ser definido dentro dos delimitadores de um elemento <Script>. Alternativamente, o elemento <ImportedScripts> é utilizado para importar um segmento de código de um ficheiro definido no atributo file.

5. APLICAÇÃO

No projeto MASSIVE foi desenvolvida uma série de provas de conceito, uma das quais intitulada de BikeDemo - uma descida virtual de uma montanha em bicicleta. A título de exemplo de aplicação do modelo proposto, apresentamos nesta secção um mapeamento entre os diferentes elementos desta demo - dispositivos de entrada e saída, e cena e seus elementos virtuais - com as entidades definidas no modelo. As ferramentas de autoria propriamente ditas estão ainda em desenvolvimento, pelo que parte dos componentes não é ainda importável/exportável no formato de ficheiro proposto, mas a estrutura da experiência segue o modelo proposto.

Esta experiência é bastante abrangente, tanto a nível de conteúdos, como de sensações oferecidas e mesmo de mecanismos de entrada de dados e controlo. A experiência é completamente virtual e recorre a uma bicicleta real devidamente instrumentada para permitir o controlo e movimentação por parte do utilizador (Figura 7). A velocidade é imposta pela gravidade (pedais não instrumentados), mas é controlada pelos travões; a trajetória é definida pela posição do guiador. Para a instrumentação dos travões optou-se pela utilização de um sensor de flexão, que varia a resistência conforme a dobra criada sobre ele.

```
<Experiment name="Sample">
  <!-- SECTION A -->
  <renderers>
    <SmellRenderer id="SensoryCol" />
    <VisualRenderer id="Rift1" class="Stereo3DView" />
    <AudioRenderer id="Binaural1" />
    <HapticsRenderer id="ButtKicker1" />
    <HapticsRenderer id="Vest1" />
  </renderers>

  <!-- SECTION B -->
  <inputs>
    <AudioDetection id="mic1" />
    <Tracker id="TrackIR1" class="PointTracker" />
    <LowLevelSensor id="Brake1" class="IDInput" />
    <Tracker id="Rift1" class="HeadTracker" />
  </inputs>

  <!-- SECTION C -->
  <Scenes>
    <Scene id="forest">
      <Environment
        place="Mountains"
        time="Afternoon"
        weather="Rainy"
        gravity="normal"/>

      <!-- SECTION C.1 -->
      <Users>
        <User id="Biker" objectRef="BikerObj">
          <renderers>
            <RenderRef id="Rift1" />
            <RenderRef id="Binaural1" />
          </renderers>
          <inputs>
            <InputRef id="TrackIR1" />
            <InputRef id="Rift1" />
          </inputs>
        </User>
      </Users>

      <!-- SECTION C.2 -->
      <objects>
        <object id="bike">
          <Transform
            Position="10 20 30"
            Rotation="0 0 0"
            Scale="1 1 1"/>
          <components>
            <ObjectComponent id="BikeControl"
              class="ControlScript" >
              <param name="SteeringInput" type="Input"
                value="TrackIR1" />
              <param name="BrakeInput" type="Input"
                value="FlexSensor1" />
              <Script>...</Script>
            </ObjectComponent>
          </components>
          <children>
            <object id="BikerObj">
              ...
              <components>
                <MeshComponent file="Biker.obj" />
                ...
              </components>
            </object>
          </children>
        </object>
      </objects>
    </Scene>
  </Scenes>

  <!-- SECTION D -->
  <controlScripts>
    <ControlScript id="RiftTrackingScript">
      <param name="HeadTrackingInput"
        defaultValue="null" />
      <Script> ... </Script>
    </ControlScript>
    <ImportedScripts file="LibraryScript1.xml" />
  </controlScripts>
</Experiment>
```

Figura 6 – Exemplo de ficheiro XML.

Uma placa Arduino mede os valores da resistência e envia-os pela porta série para o computador que corre a experiência. A rotação do guiador foi captada com um TrackIR, que dispõe do seu próprio SDK (Software Development Kit).



Figura 7 – Bicicleta instrumentada.

No decorrer da experiência são fornecidas sensações visuais estereoscópicas através de um HMD. Este funciona também como mecanismo de entrada de dados, fornecendo a posição e rotação da cabeça do utilizador. O áudio binaural, fornecido através de auscultadores com cancelamento de ruído, permite a localização dos sons no espaço. A bicicleta está munida de um oscilador de contacto de baixa frequência que alimenta o veículo com as vibrações e impactos gerados pela física da simulação, nomeadamente ao nível da vibração causada pelo relevo do terreno. O cheiro típico do ambiente natural é também passado ao utilizador. A intensidade do vento resultante da deslocação e do ambiente natural da montanha é calculada em *software*, no entanto não temos disponível ainda o *hardware* de controlo respetivo, pelo que nesta fase apenas é utilizada uma ventoinha com intensidade fixa para simular o efeito.



Figura 8 – Vista da experiência na terceira pessoa.

Esta experiência contém apenas uma cena com as condições definidas pelo ambiente representado na tabela 1.

O terreno, árvores, cercas e vários outros conteúdos são objetos que contêm componentes como sensores de colisão, que interagem com o motor de física para não permitir que sejam trespassados pela bicicleta. As flores de lavanda, tal como as árvores estão também munidas de um componente de cheiro que se fará sentir enquanto o utilizador estiver posicionado a uma distância delas em

que a atenuação do cheiro seja insuficiente para o cancelar.

Tabela 1 - Condições definidas pelo ambiente.

Propriedade	Valor
Place	Mountain
Time	Sunset
Weather	Cloudy, Dry
Gravity	Normal

A bicicleta é também um objecto (Figura 8). Este objeto tem como descendente o objeto ciclista, que além de ser visualizado é o avatar associado ao utilizador. A este utilizador estão por sua vez ligados renderizadores de imagem, áudio, cheiro e vento, que se encarregam de enviar as informações de cada sensação para o exterior. Os renderizadores trabalham em conjunção com os componentes aplicados aos objetos, sendo calculados em tempo real, por exemplo, a atenuação de cada cheiro ou som com base nas posições e outros parâmetros relativos à combinação entre renderizadores e componentes dos objetos. Na seguinte tabela apresentamos a relação entre diversos componentes e os renderizadores aplicados ao utilizador.

Tabela 2 - Componentes e respetivos renderizadores.

Renderizador	Componentes
Visual (Estereoscópico)	Meshes, Materiais (texturas/shader)
Auditivo (Binaural)	Fontes de som mono
Háptico	Sensores de colisão, Materiais (texturas de relevo)
Olfativo	Fontes de cheiro

Os materiais são conjugados com o renderizador háptico, para que a vibração e o impacto transmitidos ao dispositivo háptico dependem do relevo do solo no trajeto da bicicleta, codificado no material como um *heightmap*. Por outro lado, os sensores de colisão quando despoletados por outros objetos, enviam um sinal para o renderizador, que transmitirá impacto à bicicleta. Simultaneamente, sensores de colisão aplicados ao ciclista virtual fazem com que o renderizador háptico ative atuadores presentes num colete aplicado ao utilizador real sempre que se dá um choque entre o utilizador virtual e um objeto externo (um ramo de uma árvore, por exemplo).

A implementação da experiência foi realizada com base no modelo aqui apresentado, havendo por isso uma separação entre o *hardware* propriamente dito e a descrição dos estímulos (através dos renderizadores. Como exemplo, o projeto MASSIVE dispõe de três dispositivos diferentes para a emissão de cheiro: Scentroid, SensoryCo e um dispositivo genérico. O renderizador encarrega-se de redirecionar a saída para a máquina relevante.

O vento ambiental é renderizado com base nos parâmetros do ambiente, definidos na cena e não no utilizador. Por outro lado, a deslocação de ar sentida com a deslocação da bicicleta será calculada em tempo real. Para situações deste tipo, mais específicas, é utilizado o componente de scripting, que permite estender as propriedades e comportamentos dos objetos. O componente de scripting é também utilizado neste caso para filtrar o controlo da bicicleta. Por exemplo, ao virar o guiador, tanto o eixo frontal da bicicleta (também organizado numa estrutura hierárquica como filho do objeto bicicleta) como as mãos e braços do ciclista são alterados. Este é o tipo de casos em que o componente de *scripting* é útil: situações que necessitam de comportamentos bastante específicos. Será também uma forma de controlar objetos com base em entradas e complementa a simplicidade do modelo apresentado com a versatilidade quando necessária.

6. CONCLUSÕES

Neste artigo apresentámos um modelo de descrição de experiências multissensoriais com suporte para vários utilizadores. Da análise do trabalho relacionado, constatou-se o interesse e necessidade deste tipo de descrições, mas verificou-se que grande parte dos trabalhos se foca na componente dos dispositivos de interação (entrada) ou na componente do controlo dos dispositivos de estímulos (saída), e sem integrar estes componentes com suporte para vários utilizadores. Numa outra vertente, as arquiteturas baseadas em componentes utilizadas em vários dos atuais motores de jogo são bastante versáteis, mas não contemplam por princípio componentes multissensoriais que permitam a fácil criação desse tipo de experiências.

O modelo apresentado cobre estas componentes, mantendo uma camada de abstração que permite separar a descrição dos efeitos sensoriais a simular dos estímulos efetivamente transmitidos ao utilizador. Estes últimos são calculados em tempo real em função das especificações constantes da descrição da experiência, para a qual propomos também um formato de ficheiro baseado em XML. Finalmente, fornecemos um exemplo de aplicação do modelo proposto, em que se exemplificam os mapeamentos possíveis entre dispositivos, elementos de cena e utilizadores.

Espera-se que o modelo sirva como suporte para a criação e entrega de conteúdos multissensoriais, nomeadamente através da criação de ferramentas de autoria, que estão atualmente a ser desenvolvidas no contexto do projeto MASSIVE, no qual este trabalho se insere.

Em termos de trabalho futuro, pretende-se testar a *framework* através de um interpretador que está a ser desenvolvido para o efeito que permitirá compilar os dados de cenários criados por utilizadores. Pretende-se testar a ferramenta com utilizadores em contexto real como por exemplo profissionais em áreas como a psicologia ou a arquitetura e aferir resultados decorrentes das tarefas. Julgamos que será possível, após a otimização das ferramentas propostas, um elevado grau de autonomia por parte dos utilizadores no processo de autoria de cenários multissensoriais virtuais.

7. AGRADECIMENTOS

Este trabalho foi financiado pelo governo português através da FCT - Fundação para a Ciência e a Tecnologia e pela União Europeia (COMPETE, QREN E FEDER) no âmbito do projecto REC I/EEI-SII/0360/2012, com a designação “MASSIVE - Multimodal Acknowledgeable multiSenSorial Immersive Virtual Enviroments”.

8. REFERÊNCIAS

- [1] Oviatt, S., Cohen, P., Wu, L., Duncan, L., Suhm, B., Bers, J.,...Ferro, D. (2000). Designing the User Interface for Multimodal Speech and Pen-Based Gesture Applications: State-of-the-Art Systems and Future Research Directions. *Human-Computer Interaction*. doi:10.1207/S15327051HC11504_1
- [2] Irawati, S., Calderón, D., & Ko, H. (2005). Semantic 3D object manipulation using object ontology in multimodal interaction framework. *Proceedings of the 2005 International Conference on Augmented Tele-Existence*, 35–39.
- [3] Ob, S., & Hahn, M. (2010). The ontology for experience media authoring system. *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010, 1, 207–210*. doi:10.1109/ICCSIT.2010.5564083
- [4] Gutierrez, M., Thalmann, D., & Vexo, F. (2005). Semantic virtual environments with adaptive multimodal interfaces. *Proceedings of the 11th International Multimedia Modelling Conference, MMM 2005*, 277–283. doi:10.1109/MMMC.2005.65
- [5] B. Xiao, R. Lunsford, R. Coulston, M. Wesson, and S. Oviatt. Modeling multimodal integration patterns and performance in seniors: toward adaptive processing of individual differences. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 265–272. ACM Press, 2003.
- [6] M. Waltl, C. Timmerer, H. Hellwagner, “A Test-Bed for Quality of Multimedia Experience Evaluation of Sensory Effects”, *Proceedings of the First International Workshop on Quality of Multimedia Experience (QoMEX 2009)*, San Diego, USA, July, 2009.
- [7] Timmerer, C., Waltl, M., Rainer, B., & Hellwagner, H. (2012). Assessing the quality of sensory experience for multimedia presentations. *Signal Processing: Image Communication*, 27(8), 909–916. doi:10.1016/j.image.2012.01.016
- [8] Waltl, M., Rainer, B., Timmerer, C., & Hellwagner, H. (2013). An end-to-end tool chain for Sensory Experience based on MPEG-V. *Signal Processing: Image Communication*, 28(2), 136–150. doi:10.1016/j.image.2012.10.009
- [9] C. Timmerer. A Summary of the Representation of Sensory Effects(RoSE). 2008.

Interactive configurable virtual environment with Kinect navigation and interaction

João Pinto¹
jhpinto@ua.pt

Paulo Dias^{1,2}
paulo.dias@ua.pt

Sérgio Eliseu³
s.eliseu@ua.pt

Beatriz Sousa Santos^{1,2}
bss@ua.pt

¹DETI/UA - Department of Electronics, Telecommunications and Informatics

²IEETA - Institute of Electronics and Telematics Engineering of Aveiro

³iD+ / i2ADS - Faculty of Fine Arts, University of Porto

Abstract

As a solution to immersive virtual museum visits, we propose an extension upon the platform we previously developed for Setting-up Interactive Virtual Environments (pSIVE) that maintains all of the Virtual Environment creation features of the platform as well as content association (PDF, Video, Text), but also allows gesture-based interaction and one to one navigational input using skeleton tracking with a Kinect that is calibrated in the real world space where the user is located in. In order to validate our new navigation and interaction methods, a comparative study was performed, comparing our gesture-based interaction and positional tracking with a controller button navigation and interaction method.

Keywords

Virtual Environment, Navigation and Interaction, User Study, Kinect, Virtual Reality

1. Introduction

With the rising popularity of high-end Virtual Reality (VR) hardware in the entertainment market, immersive VR applications start to have wider expression due to cost reduction and software availability.

In 2013, as a response to the complexity of building a virtual environment, the platform for Setting up Interactive Virtual Environments (pSIVE) was developed as a master thesis [Souza 13] to not only allow an easy configuration of a virtual scene using a diversity of models, but also interact with the environment using non-conventional hardware such as trackers and head-mounted displays (HMDs).

Within that context, and as an expansion upon that project, we have added the possibility of exploring a fully immersive environment with one to one position mapping from the real world to the virtual world, as well as gesture-based interaction with menus and content within that virtual world detected by a Kinect. What this means is that the user can configure a virtual museum with custom content and navigate through it either hands-free by walking in a real empty room, as shown in figure 1, or standing still with a physical controller, while viewing the virtual museum through an HMD.

In this paper, we discuss the platform's architecture, its fea-

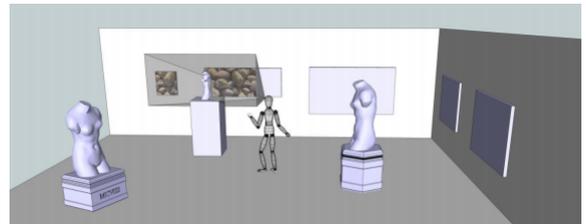


Figure 1: Virtual Museum concept

tures, how it is configured and the ways in which the users can interact with the virtual content (menus and navigation).

2. Related and Previous Work

One of the most commonly used types of display in Virtual Reality applications is the Cave Automatic Virtual Environments (CAVE) [Burdea 03], shown in figure 2. It consists of a room where the entire surface area is covered with high resolution projections that generate rapidly alternating images for each eye, allowing any number of users in the room to perceive the visual data. Stereoscopic shutter glasses are synchronized with the projectors in order for the user to only see the images for the correct eye, which enables depth perception. The projectors are placed out-

side of the room, and are controlled by one or more computers.



Figure 2: Multiple users in a CAVE [Craig 09]

Our work possesses some similarities to a CAVE system, in regards to the usage of the physical space for navigation. While in a cave the users are restricted to the empty room they are situated in, with the walls acting as a "screen" of sorts, our work is adaptable to any room, and utilizes an HMD as the user's window into the virtual world.

The Empty Museum [Hernandez 03] is perhaps the application in literature that most closely resembles ours, thematically and practically. It features a multi-user immersive walkable and wireless system where users can navigate a virtual environment with content such as audio, text, images and video. The system set up (figure 3) consists of a laptop computer in a backpack that renders the environment according to the user's perspective, connected to a system that captures wireless movement [Foxlin 98] and using virtual reality goggles.



Figure 3: Empty museum setup [Hernandez 03]

When compared to our system, The Empty Museum has the advantage of supporting multiple users at the same time, with 3D avatars within the virtual environment. Its interaction, however, is solely determined by the user's position, while we support button input for interaction, as well as gestures, allowing for more complex interactions. Their tracking system must also be mounted on the ceiling, while ours is easy to mount and easily expandable by setting up more kinects.

Brown University's VENLab [Tarr 02] is an immersive virtual reality space, with a 12m² walkable area that uses an IS-900 head tracker to measure the user's position in real

time, and an 80 degree field of view HMD to fully immerse the user in the virtual environment (figure 4).

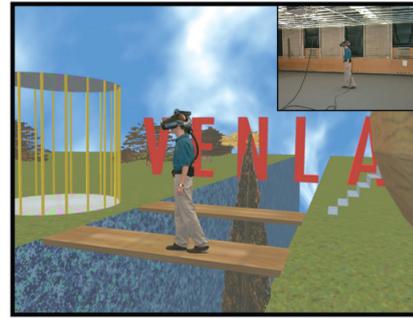


Figure 4: User walking in the VENLab space [Tarr 02]

The KidsRoom's (figure 5) [Bobick 99] aim is to explore computer vision technologies in an augmented reality interactive playspace for children. To do so, it recreates a child's bedroom composed of two real walls and two large video projection screens where images are projected from outside of the room, as well as ceiling mounted colored lights, speakers, cameras and a microphone, all controlled by a six-computer cluster. There are 4 cameras pointing at the room, one for tracking people, two for action recognition of people in the room, and one to provide a view of the room for spectators.



Figure 5: KidsRoom bedroom setup [Bobick 99]

pSIVE, a platform we previously developed, can be used to set up a virtual scene using a diversity of models, and those models can have a variety of content (PDF, Video, Text) attached to them. It uses OpenSceneGraph [Wang 10] as a graphical engine and VRJuggler [Bierbaum 01] as a middleware to interpret input from trackers and controllers, which are used to define orientation within the scene and interact/navigate, respectively.

Content can be accessed through a 2D linear menu that pops up when the user presses a button on his controller while looking at an object that has been configured with content.

3. Platform

Taking advantage of the previous work done in pSIVE, and still with its goals of easy configuration of the scene, support for content such as videos, PDF, text and images, and versatility of hardware in mind, we have expanded and up-

dated upon that platform. Support for the Oculus Rift¹ was added, as well as abandoning the VRJuggler library used by pSIVE, which we found hard to set-up and hasn't been updated since 2013, in favor of osgVRPN², a Virtual Reality Periphery Network (VRPN) [Taylor 01] library for osg that gives us lower level control of the input into our system in addition to being easier to configure.

VRPN is a library that provides a network-transparent interface between applications and physical devices used in VR systems using a client-server approach (the client being the application, and the server being the interpreter of input from the physical devices). It provides a layer of abstraction that classifies inputs into several categories (Analog, Button, Dial, ForceDevice, Sound, Text, and Tracker), which allows us to receive generic input from different devices.

With these tools and features in mind, we have designed a platform that lets the user experience a personalized virtual museum, complete with interactive content accessed through menus using gestures or a physical controller.

In the ideal use case scenario, the user carries a laptop connected to an Oculus Rift and running the client application on a backpack, while one or more computers run servers sending, to the client computer, the user's skeleton data from kinects, calibrated to the empty room the user is in.

Another use case scenario lets the user navigate using a physical controller, such as a WiiMote, removing the need to physically navigate in an empty room and carry a laptop on his/her back.

3.1. System Architecture

For the graphical engine, we have decided to continue using OpenSceneGraph (used in pSIVE), due to previous work, its active community, and the availability of VR libraries such as osgVRPN, and osgOculusViewer³ (oculus rift support). Using a well-known game engine such as Unity⁴ would have been a plausible alternative, if not for the pay-wall behind some features and the higher degree of control and flexibility that a more generic open source graphics engine such as OpenSceneGraph can provide.

The application's architecture and workflow is detailed in figure 6. It is configured with several XML files and receives input information from one or more VRPN Servers, which is then interpreted by the osgVRPN library. That information is then handled in one of two ways: interaction with menus or content (Menu Handler), or navigation (Camera Handler). The scene is then rendered for use with the Oculus Rift using the osgOculusViewer library.

In order to provide a fully immersive VR experience with skeleton tracked navigation in mind, the application needed to be designed to be easily expandable in terms of area covered by the Kinect⁵ sensor(s). As such, we

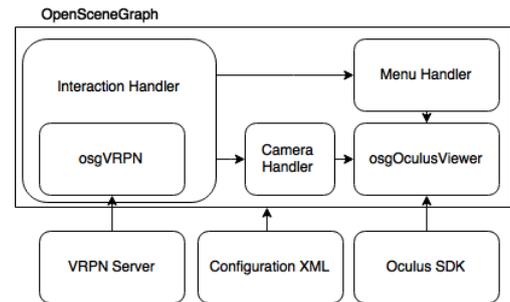


Figure 6: Application Architecture

have decided to use a PC based client-server architecture in which the client is responsible for all of the rendering of the virtual world and handling of the Head Mounted Display (HMD) orientation tracking, and the VRPN servers are used to communicate the user's skeleton information (positional data of head, hands and gripping gestures) to the client using one or several Microsoft Kinect devices collecting skeleton data with the Kinect SDK 1.8⁶. This architecture is shown in figure 7.

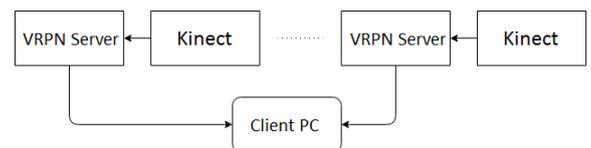


Figure 7: Client Server Architecture

osgVRPN is used in our client application with two purposes:

- Receiving information from any trackers supported by a standard VRPN server (WiiMote, Razer Hydra, Keyboard, etc.).
- Receiving information from our custom VRPN server with integrated Kinect support, something that was not available in pSIVE.

3.2. Platform Configuration

The platform, composed by the client which renders the scene and tracks the user's head orientation, and the server(s) which track(s) the user's body or read input from a physical controller, are configured through 3 XML files.

- Config.xml configures the scene, its objects and the object's contents with configurable scale, position and rotation within the scene.
- Kinect.xml configures the custom VRPN Kinect servers to listen to, using their server name and IP address.
- Controls.xml configures the navigation input buttons, allowing configuration of directional input and menu activation input.

¹<https://www.oculus.com/>

²<https://github.com/VirtualMe/osgvrpn>

³<https://github.com/bjornblissing/osgoculusviewer>

⁴<http://www.unity3d.com/>

⁵<https://www.microsoft.com/en-us/Kinectforwindows/>

⁶<https://www.microsoft.com/en-us/download/details.aspx?id=40278>

3.2.1. Scene configuration

The Config.xml file consists of a list of models, their physical attributes (size, rotation, location), and their available content.

- Filename: Path to the model.
- Label: Model description that is shown when an object is interactive.
- Rotate/Translate: User defined object rotations and translations in all axis.
- Context: Whether the object is interactive or not.

In that same file, each model has a list of available content with the following properties:

- Type: From 0 to 3, Text, PDF, Video and Image data types, respectively.
- Label: Menu entry label.
- userContent: In case of the Text data type, the text to display. In all other data types, it is the path to the content to be shown.

3.2.2. Server Configuration

In order to utilize the same library that is used for the button input (osgVRPN), a custom VRPN server with 12 channels and 3 buttons was created to support the information gathered by the Microsoft Kinect sensor. These channels are, in order, the left hand's x,y,z position, the right hand x,y,z position, the head's x,y,z position and the Kinect x,y,z position within the virtual scene.

The buttons are used to convey three binary inputs:

- Left hand grip.
- Right hand grip.
- Skeleton detected.

The left and right hand grip inputs are detected using the Kinect 1.8 SDK, and are used in most of the hands-free interactions. The skeleton detection is necessary in order for the client application to gather information from the Kinects that are detecting the user among the many Kinect servers the client application can be communicating with at any given time.

All positional information is conveyed after being transformed into virtual world coordinates using the calibration method described in section 4.

4. Kinect Calibration

The servers need to be configured with a transformation matrix in order to transform the Kinect coordinates (distance in meters from the Kinect camera) into our virtual world coordinates.

Visualization ToolKit (VTK) [Schroeder 98] was used due to previously developed work, familiarity, and easy access to its Iterative Closest Point (ICP) algorithm, to develop a program that grabs a 3D cloud from a Kinect sensor (shown in figure 8) and allows the user to roughly position it within a 3D model of the room (designed using SketchUp) that the Kinect is in, using the keyboard to translate and rotate the frame (roughly positioned frame shown in figure 9).

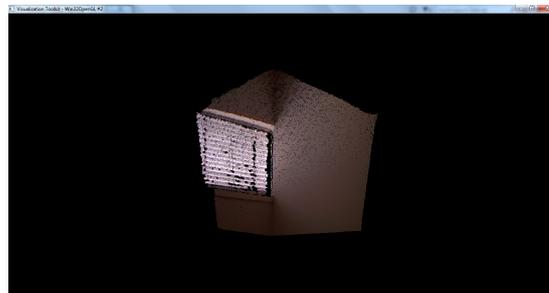


Figure 8: Captured 3D cloud

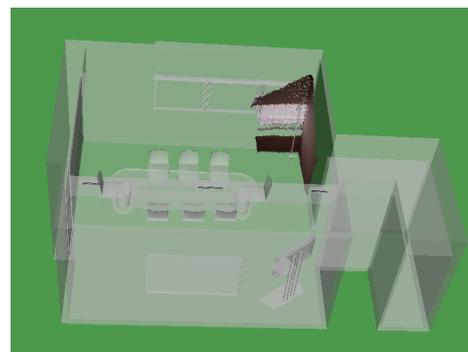


Figure 9: Manually overlaid 3D cloud

In order to fine tune the position of the frame we use the ICP algorithm [Besl 92] present in VTK, and the frame is matched as closely as possible to the 3D model (the outcome of this algorithm is shown in figure 10). The output of the ICP algorithm is a transformation matrix that we combine with the user manual positioning that when applied to the data collected from the Kinect successfully converts points from one coordinate system to the other.

That 4x4 matrix is finally exported from the calibration program as a *.mtx file, ready to be read as a command line parameter by the server that is collecting information from the Kinect from which the matrix was calculated.

5. Interaction

When developing a VR application, and with the advent of more advanced HMDs such as the Oculus Rift, VIVE and Morpheus, some of the traditional approaches to user interfaces and the display of information have to be reconsidered and at times re-purposed for use in a VR environment.

We took advantage of the use of an HMD with orientation tracking in order to utilize a gaze directed method of se-



Figure 10: 3D cloud after ICP algorithm

lecting objects within the virtual scene using a ray-tracing technique. This method was tested and compared against using a hand held selection tool with orientation and position, described as a "laser pointer selection" method in [Souza 14], and was proven to be both faster and less prone to error regardless of distance to the selected object.

Interactive objects display a configurable label in the center of the screen when looked at, which indicates that upon activation a menu containing the object's content will pop up in front of the user.

Input in our system can be done in one of two ways:

- Gestures using a Microsoft Kinect (SDK 1.8).
- Button input using any kind of controller device supported by VRPN.

5.1. Navigation

There are two ways to navigate the virtual environment, with different use cases in mind:

- Skeleton tracking based navigation using a custom VRPN server with a connected Kinect.
- Physical controller based navigation that does not require the user to move in the physical space.

Navigation in this skeleton tracking mode is done by tracking the user's head with a Kinect and positioning the camera correctly within the scene using the tracking data, and as such does not require navigational "input" per se.

It is worth noting that while the user's position in the scene is being tracked by the Kinect sensor, the HMD does not know the user's correct initial head orientation. To solve this issue, we require that the user initially performs a calibration gesture (grip with his/her right fist) while looking at the closest real-world Kinect sensor in order to offset the HMD orientation to look at the representation of that Kinect in the virtual world.

Navigational input using a physical controller is gaze-directed and reads button information from a generic VRPN server as input to our application. In order to keep the number of buttons needed to a minimum, the button to

exit a menu is the "right" button. For that same reason (the movement buttons are also used to navigate the menu system), navigation in this mode is locked when the user has a menu or content open, while in the gesture-based mode the user can move freely while browsing menus or experiencing content. Table 1 shows button and gesture mapping for navigation actions.

Action	Gesture	Button
Navigate forward	Walk	Up key
Navigate backward	Walk	Down key
Navigate left	Walk	Left key
Navigate right	Walk	Right key

Table 1: Navigation gesture/button mapping

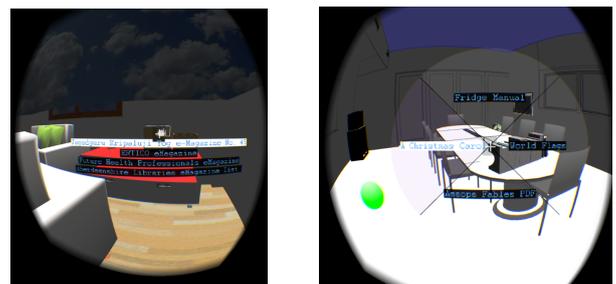
5.2. Menus

Maintaining the standard used in pSIVE, we have implemented two styles of adapted 2D menus in our application. Adapted 2D menus are a representation of standard 2 dimensional menus rendered on 3D geometry within the scene, and are the most prolific group of 3D system control techniques [Bowman 04], and can be present in multiple forms (linear, radial, etc.).

Things like the position, size and shape of menus have to be taken into special consideration due to specifications such as the fish-eye lens distortion in some HMD's, and particularly in the one used in our implementation, the Oculus Rift, which makes text hard to read when close to the edges of the screen. With those constraints in mind, we made the decision to place the menus and content centered on the user's field of view, where his/her gaze is focused, as well as avoiding the use of thin text as advised in the Oculus best practises guide [VR 15].

The application supports two different types of adapted 2D menus, illustrated in figure 11, namely linear and radial menus.

Linear menus are typically displayed when using a physical controller to interact with the scene, and are a modified version of the linear menus used in pSIVE due to an incompatibility with the previous implementation and the render distortion needed in order to use the Oculus Rift.



(a) Linear menu (Left eye) (b) Radial menu (Left eye)

Figure 11: Menu types

When interacting using gestures, radial menus are displayed. We have implemented this format for our gesture-based interaction due to faster selection times and lower error rates [Callahan 88] [Chertoff 09] when compared to linear menus.

Table 2 shows button and gesture mapping for actions related to menu interaction.

Action	Gesture	Button
Browse Menu	Move right hand	Up/Down key
Activate Option	Grip right hand	Activation key
Exit Menu	Grip both hands	Right key

Table 2: Menu browsing gesture/button mapping

5.3. Content

Content, such as images, videos (figure 12a) or PDF files (figure 12b) are rendered on a quadrangular plane and placed in front of the user. When the content is active, if pertinent, certain interactions such as browsing the PDF pages are possible.

Table 3 shows the button and gesture mapping for interaction with content.



(a) Video content (Left eye) (b) PDF content (Left eye)

Figure 12: Content rendered on a quadrangular plane

Action	Gesture	Button
Previous page	Grip left hand	Left key
Next page	Grip right hand	Right key
Exit content	Grip both hands	Down key

Table 3: Content browsing gesture/button mapping

6. Demo Environment

For the demonstration and testing environment, we modelled a meeting room in SketchUp after a real meeting room in our department, and added several objects that are not present in the real room. Each of those objects was then configured to contain various text, video, pdf and image contents.

In order to test the Kinect-based navigation method, we set up and calibrated two Kinects in the room (set up seen on

figure 13, user interaction seen on figure 14), one running on each laptop.



Figure 13: Room setup



Figure 14: User performing gesture

7. Comparison of Navigation and Interaction Methods

In order to analyze and compare the two implemented navigation and interaction techniques (Kinect-based or controller-based), we ran an experiment with 12 volunteer participants from the Aveiro University's Summer Academy, which means that participants' range of ages is not very varied, ranging from 15 to 17 years.

7.1. Methodology

This experiment aimed to verify if the two methods of interaction and navigation are equally usable in our demonstration environment. Our experiment had two input variables, namely the two different navigation and interaction methods. In one method, the users walk in the room to navigate, and use gestures to interact with the scene. In the other method, the user utilizes a controller with buttons to navigate and interact. Both methods display radial menus. The output variables of our experiment are given in measures of time and number of mistakes made.

The experiment consisted of a simple test of navigation and interaction: The participants had to navigate from one end of the virtual room to the other, and interact with a plant pot located on a table (shown in figure 15). Upon interaction, a menu is shown, and the goal is for the users to select the "Flower" option available on the menu.

Participants were given an ID at the start of the experiment (12 users, IDs ranging from 0 to 11). Even numbered participants ran the experiment using the controller first, and odd numbered participants used the Kinect first, in order to attenuate possible bias due to learning effects. The participants were given no time to practice and learn the system beforehand. The participants were observed while performing the experiment and were asked to answer a questionnaire regarding their satisfaction, difficulties and preferred method. It is worth noting that while 12 participants ran the experiment, only 11 delivered the questionnaire.

Measurements were taken automatically during the experiment, and consisted of the time the participant took to get in front of the plant pot and activate the menu, the time they took to select the correct option, as well as any incorrect options selected.

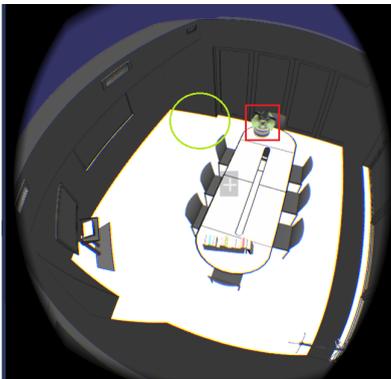


Figure 15: Experiment environment (interaction zone circled)

7.2. Results

Regarding the times measured, the differences in the time taken to reach the plant pot are negligible with the controller-based method averaging 35,25 seconds to reach the plant pot versus the 35,41 seconds of the Kinect-based method while the interaction was faster when using the controller-based method, which averaged 16,6 seconds to activate the correct option versus the 30,3s that the users took with the Kinect.

Users also made more mistakes with the Kinect-based method, with three participants out of twelve making one mistake in selecting the right option, while nobody selected the wrong option using the controller.

The questionnaire results are detailed in tables 4, 5 and 6. Tables 4 and 5 indicate participant answers to several parameters, represented as a median of each index in a scale of 1 (disagree) to 5 (agree), in regards to navigation and interaction, respectively. Table 6 details the number of participants that claim to prefer one method or the other, or have no preference.

While tables 4 and 5 seem to indicate little difference between the two options, the Kinect-based method seems to be predominantly preferred in both navigation and interaction, as shown in table 6.

	Kinect	Controller
Ease of positioning	4	4
Intuitiveness	4	4
Has Annoying features	2	2
Requires training	5	5
Satisfaction	4	4

Table 4: Questionnaire results regarding navigation

	Kinect	Controller
Ease of positioning	4	4
Intuitiveness	4	4
Has Annoying features	3	2
Requires training	5	3
Satisfaction	4	4

Table 5: Questionnaire results regarding interaction

	Navigation	Interaction
Kinect	7	6
Controller	1	3
Indifferent	3	2

Table 6: Questionnaire results regarding preferences (in number of participants)

With all this in mind, our results seem to show that even though the Kinect-based method for navigation and interaction is slower on both accounts, participants seem to prefer it, possibly due to the novelty factor.

8. Conclusions and future work

This paper presents an evolution and rework of pSIVE, with the goal of providing the tools to enable immersive museum visits in virtual reality. While pSIVE allowed for easy creation of Virtual Environments without knowledge of any programming languages, it lacked the tools for real time positional and gesture tracking with a Kinect, as well as the calibration tools necessary to tie the virtual world coordinates to real world positions. We have also implemented a new style of menu (radial), to accompany our implementation of gesture-based interaction.

The results of our experiment with participants seem to indicate a positive reaction, with the majority of users preferring to use the Kinect-based method of navigation and interaction within the virtual environment. Navigation wise, both methods seem to perform equally. When it comes to interaction, gestures seem to be fairly slower than button input. A proposed solution to make selection times faster is the implementation of a gaze directed method with gesture actuation selection.

Future work can involve multiple aspects, from implementing new and more efficient styles of menus and interaction methods, to finding a way of automating the calibration process. An expansion upon this project is already

in the works, with plans to allow users to create their own museum from a list of 3D models and using gestures to place them in the environment, in real time, as well as saving and sharing their custom museum with other users.

Acknowledgements

This work is supported by National Funds through FCT - Foundation for Science and Technology, in the context of the projects UID/CEC/00127/2013 and Incentivo/EEI/UI0127/2014.

References

- [Besl 92] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, February 1992.
- [Bierbaum 01] Allen Bierbaum, Christopher Just, Patrick Hartling, Kevin Meinert, Albert Baker, and Carolina Cruz-Neira. Vr juggler: A virtual platform for virtual reality application development. In *Proceedings of the Virtual Reality 2001 Conference (VR'01)*, VR '01, pages 89–, Washington, DC, USA, 2001. IEEE Computer Society.
- [Bobick 99] Aaron F. Bobick, Stephen S. Intille, James W. Davis, Freedom Baird, Claudio S. Pinhanez, Lee W. Campbell, Yuri A. Ivanov, Arjan Schütte, and Andrew Wilson. The KidsRoom: A Perceptually-Based Interactive and Immersive Story Environment, 1999.
- [Bowman 04] Doug A. Bowman, Ernst Kruijff, Joseph J. LaViola, and Ivan Poupyrev. *3D User Interfaces: Theory and Practice*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 2004.
- [Burdea 03] G. Burdea and P. Coiffet. Virtual reality technology. In *Presence: Teleoperators & Virtual Environments*, volume 12, pages 663–664, 2003.
- [Callahan 88] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman. An empirical comparison of pie vs. linear menus. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '88*, pages 95–100, New York, New York, USA, May 1988. ACM Press.
- [Chertoff 09] Dustin B. Chertoff, Ross Byers, and Joseph J. LaViola. Poster: Evaluation of menu techniques using a 3D game input device. In *2009 IEEE Symposium on 3D User Interfaces*, pages 139–140. IEEE, 2009.
- [Craig 09] Alan B. Craig, William R. Sherman, and Jeffrey D. Will. *Developing Virtual Reality Applications: Foundations of Effective Design*. Morgan Kaufmann, 2009.
- [Foxlin 98] Eric Foxlin, Michael Harrington, and George Pfeifer. Constellation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques - SIGGRAPH '98*, pages 371–378, New York, New York, USA, July 1998. ACM Press.
- [Hernandez 03] L. Hernandez, J. Taibo, A. Seoane, R. Lopez, and Rocio. Lopez. The empty museum. Multi-user interaction in an immersive and physically walkable VR space. In *Proceedings. 2003 International Conference on Cyberworlds*, pages 446–452. IEEE Comput. Soc, 2003.
- [Schroeder 98] Will Schroeder, Kenneth M. Martin, and William E. Lorensen. *The Visualization Toolkit (2Nd Ed.): An Object-oriented Approach to 3D Graphics*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1998.
- [Souza 13] Danilo Souza. Platform for Setting up Interactive Virtual Environments, 2013.
- [Souza 14] Danilo Souza, Paulo Dias, and Beatriz Sousa Santos. Choosing a selection technique for a virtual environment. In *Lecture Notes in Computer Science*, volume 8525 LNCS, pages 215–225. Springer Verlag, 2014.
- [Tarr 02] Michael J Tarr and William H Warren. Virtual reality in behavioral neuroscience and beyond. *Nature neuroscience*, 5 Suppl:1089–92, November 2002.
- [Taylor 01] Russell M. Taylor, II, Thomas C. Hudson, Adam Seeger, Hans Weber, Jeffrey Juliano, and Aron T. Helser. Vrpn: A device-independent, network-transparent vr peripheral system. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST '01*, pages 55–61, New York, NY, USA, 2001. ACM.
- [VR 15] Oculus VR. Best practices guide, January 2015. http://static.oculus.com/sdk-downloads/documents/Oculus_Best_Practices_Guide.pdf.
- [Wang 10] Rui Wang and Xuelei Qian. *OpenScene-Graph 3.0: Beginner's Guide*. Packt Publishing, 2010.

Visualização de Objetos *Off-Screen* em Realidade Aumentada Móvel

Gonçalo Silva¹Maria Beatriz Carmo¹Ana Paula Afonso²Ana Paula Cláudio¹António Ferreira²¹ BiolSI, Biosystems & Integrative Sciences Institute, Faculdade de Ciências da Universidade de Lisboa² LaSIGE, Faculdade de Ciências, Universidade de Lisboa

Campo Grande, 1749-016 Lisboa, Portugal

fc45337@alunos.fc.ul.pt, {mbcarmo, apafonso, apclaudio, amferreira}@fc.ul.pt

Abstract

Augmented Reality (AR) on mobile devices is an expanding area. In the last decade, we have seen a huge growth in the development of applications that use this technology to add useful information to the user concerning the environment that surrounds him. However, the use of AR in small devices is challenging. For example, when visualizing points of interest, there is no prior control over their number and location, thus, the overlapping of several symbols may occur. On the other hand, the addition of information to the real image does not have to be confined to objects in the user's field of view. The inclusion of clues about the existence of relevant objects outside the field of view (off-screen objects) is an important contribution for navigation through the space of information available to the user. Another aspect to keep in mind is the ability to express the information's relevance, this is, adapt the representation according to user's preferences. This paper presents a prototype developed for Android that proposes new techniques to reduce symbol occlusion, represent object relevance, and include cues for off-screen objects in mobile augmented reality environments.

Keywords

Augmented reality, mobile devices, off-screen objects, relevance, overlapping.

Sumário

A realidade aumentada (RA) em dispositivos móveis é uma área em crescente expansão. Na última década tem-se assistido a um desenvolvimento acentuado de aplicações que fazem uso desta tecnologia para acrescentar informação útil ao utilizador relativamente ao ambiente que o rodeia. No entanto, o uso da RA em dispositivos de pequenas dimensões coloca alguns desafios. Por exemplo, ao visualizar pontos de interesse, não existe controlo prévio sobre o seu número e localização, podendo ocorrer a sobreposição de vários símbolos. Por outro lado, a adição de informação à imagem real não tem que ficar confinada aos objetos presentes no campo de visão do utilizador. A inclusão de pistas sobre a existência de objetos fora do campo de visão (objetos *off-screen*) é um contributo importante para a navegação pelo espaço de informação. Outro aspeto a ter em conta é a capacidade de expressar a relevância da informação, ou seja, adaptar a representação de acordo com as preferências do utilizador. Este artigo apresenta um protótipo desenvolvido em Android, que propõe novas técnicas para reduzir sobreposições de símbolos, representar a relevância dos objetos, e incluir pistas para objetos *off-screen* em ambientes de realidade aumentada móvel.

Palavras-Chave

Realidade aumentada, dispositivos móveis, objetos *off-screen*, relevância, sobreposições.

1. INTRODUÇÃO

A realidade aumentada (RA) consiste em sobrepor elementos virtuais, em tempo real, à imagem do mundo físico capturada pela câmara de um dispositivo. Para uma experiência mais rica, os elementos virtuais devem movimentar-se no ecrã, acompanhando o movimento do dispositivo nas mãos do utilizador [Azuma97]. Com o crescimento massivo do uso de *smartphones* e *tablets* que integram vários sensores tais como *GPS*, acelerómetro,

magnetómetro, giroscópio, entre outros, estão reunidas as condições para o desenvolvimento de aplicações de RA para estes dispositivos. Um exemplo deste padrão emergente consiste no aumento da popularidade de aplicações na área do turismo, que sinalizam pontos de interesse em dispositivos móveis, tais como a Layar¹ ou a

¹ <https://www.layar.com/>

Wikitude². Estas aplicações, para além de mostrarem pontos de interesse num ambiente de RA, são usadas para obter variadas informações através de elementos textuais e visuais colocados sobre esses mesmos pontos [Pence10]. Um dos desafios destas aplicações é mostrar a informação de forma adequada, permitindo uma análise e compreensão que estimulem uma reação apropriada e intuitiva por parte do utilizador que a observa.

No entanto, quando as aplicações de RA são usadas em dispositivos móveis, apresentam várias restrições. Um dos principais problemas é que na RA o utilizador está limitado ao campo de visão da câmara, mas a informação encontra-se 360º à sua volta [Mulloni10]. Assim, a informação fora do campo de visão da câmara não é visível no ecrã, limitando drasticamente a perceção do utilizador sobre aquilo que o rodeia, obrigando-o a rodar o dispositivo verticalmente e horizontalmente até encontrar os objetos pretendidos. De forma a mitigar este problema, é necessário conseguir fornecer informação no ecrã relativa à existência de objetos fora da área visível (objetos *off-screen*).

Outra restrição dos dispositivos móveis é o ecrã pequeno, que causa frequentes sobreposições entre as representações dos objetos, que se traduzem em problemas de navegação e de perceção da informação mostrada [Nóbrega15].

Por fim, outro tópico abordado é como apresentar a informação para que o utilizador identifique o que é relevante de acordo com as suas preferências, permitindo que a sua tarefa de navegação seja mais rápida e intuitiva, e o seu objetivo seja concluído com menos esforço.

Como resposta aos problemas expostos, neste artigo é descrito o protótipo de uma aplicação para o sistema operativo Android, que serve para visualizar pontos de interesse em RA móvel, explorando a representação de pontos *off-screen*, a sua relevância, e o tratamento de sobreposições.

Este artigo está organizado da seguinte forma: na próxima secção é identificado o principal trabalho relacionado, na secção 3 é apresentado o protótipo desenvolvido e as suas principais funcionalidades, e a secção 4 é dedicada às conclusões e trabalho futuro.

2. TRABALHO RELACIONADO

Existem já algumas técnicas que fazem a sinalização de pontos *off-screen* em RA móvel, apesar de nenhuma propor métodos de representação de relevância e tratamento de sobreposições. Nesta secção apresentam-se as principais técnicas que inspiraram o protótipo desenvolvido.

2.1 Radar

A técnica designada por Radar segue uma estratégia *overview+detail*, e é frequentemente usada em RA. Fornece uma vista extra (*overview*) representativa do mundo real, visto de cima, estando o utilizador no centro,

² <https://www.wikitude.com/>

o que permite perceber claramente onde se encontra e que objetos o rodeiam [Schinke10].

No entanto, esta técnica exige uma constante conversão da vista 2D apresentada para a vista real 3D, o que apesar de ser um processo mental automático dos utilizadores, pode causar alguma demora e confusão quando usado com um grande conjunto de pontos. Além disto, a limitação desta técnica para representar alturas torna desconfortável a sua utilização, pois não sendo possível saber a que altura está um objeto, o utilizador é obrigado a rodar o dispositivo verticalmente até encontrar o ponto pretendido [Aretz92].

Ainda assim, esta aproximação continua a ser uma técnica largamente utilizada, principalmente em cenários virtuais, pela sua simplicidade e eficácia em ambientes com um número de objetos limitado.

2.2 Setas 3D

De forma a resolver os problemas associados ao Radar, as setas 3D têm sido frequentemente utilizadas como meio de sinalização *off-screen* [Chitarro04]. Esta técnica consiste na utilização de setas desenhadas tridimensionalmente, que apontam para objetos *off-screen* [Burigat07].

Apesar de solucionar o problema da representação de alturas e não ser necessária qualquer conversão mental, visto que tanto o mundo físico como as setas estão representados tridimensionalmente, quando utilizada com um grande conjunto de pontos esta técnica apresenta níveis de oclusão elevados. Assim, torna-se muito difícil seguir as indicações fornecidas, pois a sobreposição entre setas torna-as ilegíveis [Chitarro04].

Foram propostas algumas alternativas, nomeadamente, a diminuição do tamanho das setas de forma a reduzir a probabilidade de interseção. Todavia, neste caso perde-se a informação dada através da morfologia da seta, por exemplo, no caso da distância do utilizador ao ponto ser representada pelo comprimento da seta.

Outro problema é o facto da vista fornecida pela técnica estar em conflito com o eixo de referência humano, o que pode tornar difícil seguir o movimento proposto pelas setas e até confundir o utilizador relativamente à direção indicada pelas mesmas [Carlson-Radvansky93].

2.3 AroundPlot

A técnica AroundPlot, proposta por Jo *et al.* [Jo11], adota uma abordagem *focus+context*, e por isso combina a vista detalhada e a vista geral numa só, o que contribui para a fácil comparação entre as vistas em tarefas de movimento e mudança de direção. A vista detalhada corresponde ao mundo real captado pela câmara do dispositivo, enquanto a vista de contexto ocupa uma moldura junto à borda do ecrã. Esta moldura é transparente e limitada por um retângulo que envolve a zona central da imagem. Os objetos *off-screen* são representados por pequenos círculos desenhados na moldura, a uma distância do bordo que é proporcional à distância do objeto à região visível na imagem.

Uma das principais vantagens desta técnica é a região da moldura onde aparece um ponto *off-screen* indicar o

sentido da rotação que deve ser efetuada para que este se torne visível (*on-screen*), induzindo um movimento intuitivo, claro e completamente alinhado com o eixo de referência humano ao contrário das setas 3D.

Por outro lado, em comparação com o radar, permite representar a altura através das regiões horizontais existentes em cima e em baixo, no ecrã. Por fim, como estas barras são demasiado estreitas para representarem uma parte tão grande do mundo real (tudo o que não é visível), surge uma funcionalidade de ampliação dinâmica das mesmas consoante o movimento do utilizador. Se o utilizador quer que um ponto colocado na barra direita se torne visível, este terá de rodar para a direita, e nesse momento, a região direita será ampliada (ampliação dinâmica) para que os pontos se possam dispersar mais livremente, e simultaneamente, proporcionando uma ideia mais precisa de quando o ponto se irá tornar visível.

Contudo, quando esta técnica é usada com muitos pontos, a moldura fica demasiado preenchida o que torna a leitura da informação sobre os objetos *off-screen* confusa e, ainda que a ampliação dinâmica reduza grande parte desse problema, os cantos continuam a ser uma zona com elevada densidade de pontos, porque pertencem simultaneamente a uma região horizontal e a uma região vertical.

2.4 SideARs

O SideARs foi proposto por Siu *et al.* [Siu13] com o intuito de fornecer suporte e apoio aos bombeiros que possuem fracas condições de comunicação, tentando aumentar a sua perceção relativamente aos recursos que os rodeiam. Os autores desta técnica fazem referência a todas as anteriormente mencionadas, afirmando que nenhuma delas faz agrupamento de pontos de interesse por tipo. O SideARs surge na tentativa de colmatar essa falha.

O protótipo do SideARs usa duas regiões retangulares verticais, uma de cada lado do ecrã, contendo vários símbolos representativos dos objetos *off-screen*, semelhante à técnica do AroundPlot. Porém, existe a funcionalidade de agrupamento de pontos consoante o tipo, como camiões, estações de polícia, entre outros. Cada agrupamento é representado por um símbolo que mostra um número indicativo do número de objetos existentes daquele tipo, e a distância até chegar ao mais próximo.

Por fim, este protótipo dá ainda a possibilidade de configurar quais os tipos de pontos que devem estar visíveis na aplicação, pois o que é relevante para um bombeiro pode não ser para outro, dependendo da sua função. Desta forma, para além de fazer a sinalização *off-screen*, o SideARs permite aos bombeiros filtrar e encontrar os objetos relevantes com maior rapidez e eficiência.

Contudo, os autores não fazem qualquer referência à representação de alturas e as indicações *off-screen* não fornecem nenhuma informação sobre quando é que o objeto vai entrar na área visível, tornando impossível

para o utilizador antecipar esse momento e abrandar a rotação.

2.5 Sparkle

Desenvolvido por Boll *et al.* [Boll14], o Sparkle, apesar de não ser usado em RA, é aqui abordado por usar a região envolvente da imagem para a sinalização de objetos *off-screen*. Esta técnica foca-se no uso de uma superfície de LEDs que emite luz, que permite indicar a localização de pontos *off-screen* sem a necessidade de colocar indicações no ecrã. Ou seja, até agora todas as técnicas de sinalização de pontos fora da área visível colocavam pistas no próprio ecrã para os encontrar, ao contrário do Sparkle, que mantém o ecrã totalmente disponível para pontos *on-screen* e usa uma superfície luminosa exterior para a sinalização *off-screen*, o que é inovador.

Para conseguir este efeito, o dispositivo é colocado sob uma superfície de LEDs emissores de luz de diferentes cores, intensidades e formas, tornando possível ao utilizador perceber onde estão os pontos *off-screen*. Esta técnica provou ser robusta, diminuindo a carga cognitiva e aumentando a perceção sobre o que rodeia o utilizador de forma intuitiva, tendo obtido bons resultados.

Porém, é uma técnica limitada na medida em que implica ter um suporte de LEDs para usar em conjunto com o dispositivo, sendo apenas viável em situações muito concretas e não no dia-a-dia do utilizador comum.

3. EXPLORAÇÃO DE PONTOS DE INTERESSE EM REALIDADE AUMENTADA MÓVEL

Nesta secção é descrito em detalhe de que forma o protótipo desenvolvido pretende resolver os problemas de representação de relevância, sinalização *off-screen*, e tratamento de sobreposições.

3.1 Contexto

O protótipo desenvolvido permite a visualização de pontos de interesse, no qual o utilizador pode navegar livremente num ambiente de RA usando dispositivos móveis.

Mais especificamente, o protótipo utiliza dados georreferenciados sobre caixas multibanco em Portugal, que constituem os pontos de interesse apresentados na aplicação. Pretendendo este protótipo ser uma prova de conceito, são considerados apenas três bancos: Caixa Geral de Depósitos (CGD), Banif (Ban) e Novo Banco (NB), que são suficientes para demonstrar as funcionalidades desenvolvidas.

3.2 Representação da Relevância

Apesar da evolução dos dispositivos móveis, os ecrãs continuam a impor restrições severas ao nível da visualização e interação, particularmente aquando da exploração de grandes quantidades de dados geográficos. Filtrar a informação no ecrã de acordo com a sua importância pode tornar-se demasiado complexo se os objetos mais relevantes não “saltarem à vista” comparativamente aos restantes, principalmente devido à capacidade de atenção limitada do ser humano [Swienty08]. É importante, no meio de tanta informação navegável, conseguir salientar a que é mais importante

para o utilizador, para que este possa encontrar o que pretende de forma rápida e eficiente.

Para desenvolver o protótipo, e de acordo com um estudo desenvolvido por Gonçalves *et al.* [Gonçalves11a], a cor foi o atributo escolhido para a representação da relevância. Neste estudo utilizou-se a analogia *warm-cold* em que as cores mais frias representam objetos menos relevantes, e as cores mais quentes objetos de maior relevância. Juntou-se ainda a variação de opacidade, sendo representados com maior opacidade os objetos mais relevantes e maior transparência os menos relevantes.

Dentro do mesmo grau de relevância, o grau de transparência varia também em função da distância do objeto ao observador. Assim, os pontos mais transparentes estão mais longe e os mais opacos mais perto. A Figura 1 resume a abordagem de cores e transparências adotada.

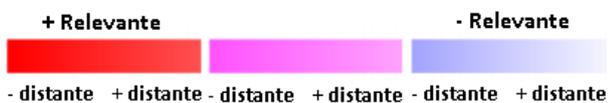


Figura 1: Esquema de cores e transparências usado no protótipo [Gonçalves11b].

Deste modo, cada símbolo representativo de um ponto de interesse possui uma cor e um grau de transparência em função da sua relevância e distância ao utilizador, respetivamente.

No protótipo aqui apresentado, a relevância foi atribuída *a priori* consoante o banco. Contudo, poderia ter-se considerado a atribuição da relevância através de uma função de grau de interesse com base em vários critérios, como sugerido por Pombinho *et al.* [Pombinho09].

Quanto à transparência, é preciso notar que, como ilustrado na Figura 1, um ponto muito relevante é sempre mais opaco que um outro pouco relevante, mesmo estando o último muito mais perto do utilizador, uma vez que o objetivo é conseguir que o utilizador se foque no que é mais importante. Por isso, a transparência apenas pode ser usada para comparar distâncias entre pontos da mesma relevância. Consequentemente, usámos a Tabela 1 como referência para implementar o mecanismo de transparência, variando esta num intervalo [0, 255], correspondendo o zero à maior transparência.

Tabela 1: Distribuição da cor e transparência de acordo com a relevância dos pontos de interesse [Gonçalves11a].

Relevância	Cor	Transparência
Elevada	Vermelho	[190, 255]
Intermédia	Roxo	[64, 189]
Baixa	Azul	[0, 63]

Para conseguir o comportamento pretendido, criou-se uma função α (1) que permite interpolar o valor da transparência em função da distância d e relevância r de

um ponto, sabendo para cada r o respetivo valor máximo e mínimo da transparência e ainda o raio de pesquisa de pontos relativamente à posição do observador:

$$\alpha(r, d) = \left(\frac{t_{\max}(r) - t_{\min}(r)}{\text{raio}} \right) \times d + t_{\min}(r) \quad (1)$$

O raio de pesquisa na função (1) é um parâmetro de configuração, que pode ser modificado pelo utilizador. A aplicação só irá considerar os POI cuja distância ao utilizador seja igual ou inferior ao valor atribuído a este parâmetro.

Para facilitar a comparação da distância dos objetos à posição do observador, além da variação do grau de transparência, como proposto por Gonçalves *et al.* [Gonçalves11a], recorreu-se também à variação do tamanho do símbolo: objetos mais distantes correspondem a símbolos mais pequenos.

Optou-se, assim, por usar dois atributos do símbolo, transparência e tamanho, para expressar a distância. Por um lado, como não é possível controlar o conteúdo da imagem do fundo, a variação do grau de transparência pode não ser perceptível, ao contrário do que acontece com a variação do tamanho. Por outro lado, como a variação da transparência é usada nos símbolos *off-screen*, como se explicará adiante, manteve-se a variação deste atributo para manter a coerência entre as duas representações.

O resultado final da combinação de cores, transparências, e tamanhos é ilustrado na Figura 2, que mostra as três cores usadas para representar relevância, respetiva transparência e variação do tamanho, consoante a distância ao utilizador.



Figura 2: Exemplo do uso de cores e transparências no protótipo.

Se o utilizador pretende comparar a distância entre objetos de diferentes relevâncias, deverá focar-se no tamanho do símbolo, que será maior quanto mais perto este estiver do utilizador. É ainda de referir que não é feita qualquer adaptação da simbologia de acordo com o ambiente exterior, por isso todos os símbolos possuem um rebordo preto, por ser uma das formas preferidas dos utilizadores distinguirem claramente os símbolos do *background*, de acordo com um estudo realizado por Carmo *et al.* [Carmo12].

3.3 Sinalização Off-Screen

Atualmente existem várias técnicas para sinalizar objetos fora da área visível no ecrã, mas muitas delas são pouco escaláveis ou confusas. Além disso, não existem técnicas que simultaneamente deem pistas para a localização dos objetos *off-screen* e informem sobre a sua relevância.

Para resolver esta limitação, seguiu-se uma abordagem *focus+context* com uma moldura de sinalização, semelhante ao *AroundPlot*. A técnica desenvolvida usa cores e transparência para representar a distância e relevância dos pontos *off-screen* e *on-screen*.

O protótipo tem ainda um modo alternativo de visualização, que complementa a RA recorrendo a um mapa 2D, para que o utilizador possa intercalar entre ambos de forma a encontrar o que pretende de forma mais rápida. Em seguida, cada um dos modos é descrito de forma detalhada.

3.3.1 Realidade Aumentada

Na camada de RA, foram criadas duas barras horizontais e duas barras verticais ligeiramente translúcidas, que formam uma moldura à volta da área visível. As duas barras que representam pontos *off-screen* na horizontal estão posicionadas nos lados esquerdo e direito do ecrã (Figura 3a), e as outras duas, que representam pontos *off-screen* na vertical, estão posicionadas nas partes superior e inferior do ecrã (Figura 3b).

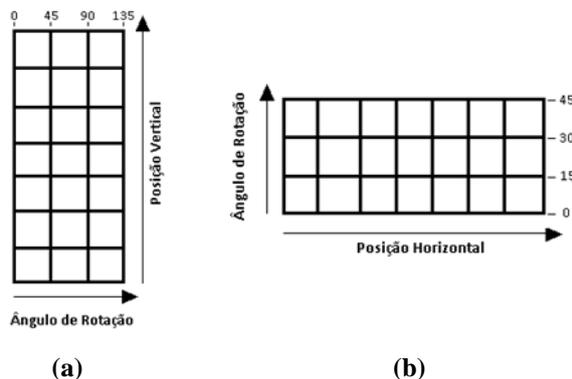


Figura 3: Barras da camada de RA: a) vertical direita, e b) horizontal superior.

De modo a evitar sobreposições, cada uma destas barras está dividida numa quadrícula para permitir agrupar os objetos *off-screen* de acordo com a sua posição no espaço.

Se um objeto está dentro do campo de visão da câmara no plano vertical, isto é, não está a uma altura demasiado elevada ou reduzida que o utilizador não o possa ver, mas está fora do campo de visão da câmara no plano horizontal (por exemplo, atrás das costas), então isso significa que o ponto aparecerá ou na barra direita ou na barra esquerda. Visto que esse mesmo ponto pode tornar-se visível rodando para a esquerda ou para a direita, o símbolo é colocado na barra que corresponde ao movimento de rotação de mais rápido acesso ao ponto.

O mesmo raciocínio é aplicado a um objeto que esteja dentro dos limites horizontais, mas fora dos limites verticais. Nesse caso, o objeto *off-screen* está a uma

altura demasiado alta ou baixa e não está dentro do ângulo de visão da câmara, pelo que deverá aparecer na barra superior ou inferior. A diferença reside em não considerar a possibilidade do utilizador rodar para além dos 180° (apontar para o céu) ou aquém dos 0° (apontar para o chão), pois seriam movimentos impossíveis para a anatomia humana e pouco expectáveis de serem realizados com o dispositivo.

O posicionamento dos símbolos representativos dos objetos *off-screen* nas barras verticais e horizontais depende da abertura da câmara e da posição do objeto relativamente à direção em que a câmara está a apontar, tanto na horizontal como na vertical. Um dos parâmetros de configuração da aplicação é o ângulo de abertura da câmara, que neste caso foi inicializado a 90°, sendo este o valor que serviu de referência aos ângulos apresentados nas Figuras 3 e 4.

Relativamente ao plano horizontal, consideraram-se 7 zonas, ilustradas na Figura 4a. Estas zonas são identificadas pelo ângulo entre a direção em que se encontra o objeto e a direção em que está apontado o dispositivo, correspondente ao ângulo 0°: a zona 0 cobre a área visível no ecrã; as zonas 1, 2 e 3 são, respetivamente, a 1ª, 2ª e 3ª colunas da barra direita, começando a contar do bordo interior da moldura; e as zonas 4, 5 e 6 são a 1ª, 2ª e 3ª colunas da barra esquerda, a contar do bordo interior da moldura.

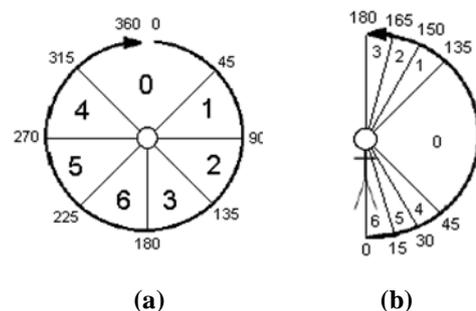


Figura 4: Rotações: a) horizontal vista de cima, e b) vertical vista de lado.

Quanto à disposição dos pontos na vertical, seguiu-se uma divisão análoga à considerada para o plano horizontal. Consideraram-se também 7 zonas, em que a zona 0 corresponde à área visível no ecrã e as restantes zonas às áreas das barras superior e inferior (Figura 4b).

Um objeto pode pertencer à interseção de duas barras. Por exemplo, se estiver simultaneamente acima e à esquerda da área visível no ecrã, pertencerá à interseção da barra superior com a barra esquerda. Tendo em atenção as áreas definidas, mostram-se na Figura 3 os limites dos ângulos de rotação a aplicar para ver no ecrã os pontos *off-screen* colocados em cada coluna e linha, respetivamente.

No protótipo construído, a determinação da célula em que é mostrado o ponto *off-screen* baseia-se no cálculo do azimute e da altitude em relação à posição do utilizador. O azimute é o ângulo formado entre o norte geográfico (*true north*) e uma outra direção, neste caso a

direção de observação do utilizador. A altitude é o ângulo formado entre o plano do horizonte e o objeto. Estes valores foram calculados usando a API Android tendo em conta os valores do GPS, magnetómetro e acelerómetro do dispositivo móvel. A partir destes valores, a posição do ponto é *on-screen* ou *off-screen* consoante o azimute (y) e altitude (p) e de acordo com os valores da função $posiçãoPonto(y,p)$ (2):

$$\begin{cases}
 \text{on-screen, se } (315 \leq y \leq 360 \vee 0 \leq y \leq 45) \\
 \quad \wedge (45 \leq p \leq 135) \\
 \text{off-screen left, se } 180 < y < 315 \\
 \text{off-screen right, se } 45 < y \leq 180 \\
 \text{off-screen top, se } p < 45 \\
 \text{off-screen bottom, se } p > 135
 \end{cases} \quad (2)$$

Uma vez determinada a posição do símbolo, este será desenhado com a cor e transparência correspondentes, respetivamente à sua relevância e distância ao utilizador (Figuras 5a e 5b).

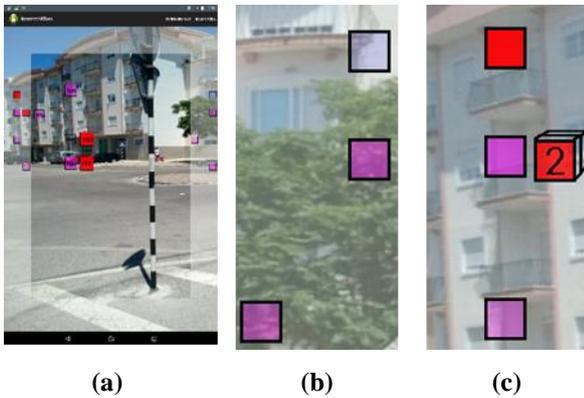


Figura 5: Sinalização *off-screen*: a) barras de sinalização, b) pontos únicos sinalizados, e c) pontos agregados sinalizados.

Resumindo, o protótipo permite ao utilizador obter as seguintes informações relativas a um ponto no ecrã: a sua relevância (dada pela cor), comparar a distância deste ao utilizador com outros pontos da mesma relevância (a partir da transparência), saber qual o lado de acesso mais rápido ao ponto (indicado pela barra onde o ponto aparece) e ainda ter noção da amplitude da rotação horizontal e/ou vertical necessária para que ele se torne visível (informação fornecida pela coluna e/ou linha onde está inserido).

Por fim, se existe mais de um ponto na mesma célula da grelha de uma das barras de sinalização, então o símbolo é desenhado com linhas adjacentes ao quadrado para dar o efeito de perspetiva, que designamos por pilha, e será mostrado um número indicativo de quantos pontos estão ali agregados. A pilha assume a cor e transparência do ponto mais relevante e mais próximo de entre todos os agregados naquela célula (Figura 5c).

3.3.2 Mapa 2D

A aplicação desenvolvida fornece uma opção de Mapa 2D, que surge como uma alternativa complementar à RA.

A RA exige que o utilizador adote uma postura vertical e segure o dispositivo firmemente de forma prolongada, o que pode causar fadiga e desconforto, visto esta ser uma posição pouco natural [Nóbrega15]. Ao fornecer o modo de Mapa 2D, permitimos que o utilizador use a RA para encontrar o que pretende e se oriente nessa direção, e que de seguida possa mudar para outro modo mais confortável, podendo voltar ao modo anterior sempre que entender.

Por outro lado, a RA pode estar a indicar pontos *on-screen* que ainda não são visíveis no mundo real, por existirem obstáculos entre o utilizador e o ponto, ou por a distância entre eles ser ainda demasiado grande. Desta forma, alternando para o Mapa, o utilizador pode situar-se e, além disso, perceber qual a posição dos pontos de interesse em relação à sua posição no terreno. O mapa, além de mostrar a localização atual do utilizador com um ponto azul, indica também os pontos que estavam na RA com a mesma cor e transparência para haver coerência nas duas representações. Por último, sobre o Mapa 2D é ainda colocado um Radar que permite visualizar o raio de pesquisa e a área *on-screen* (Figura 6).

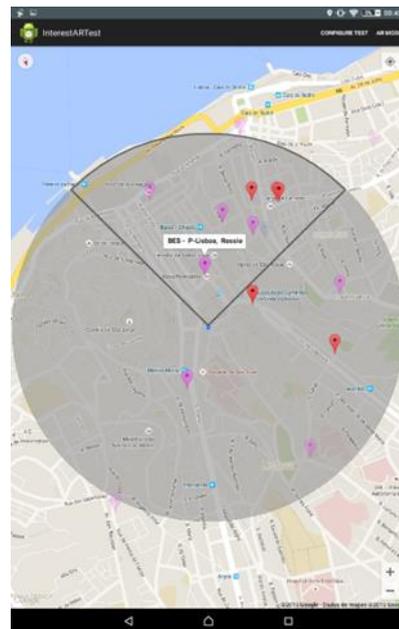


Figura 6: Modo Radar.

A aplicação tem ainda a capacidade de rodar o mapa conforme a orientação do utilizador de modo a que a orientação do mapa esteja alinhada com a do utilizador. Ao clicar sobre um ponto no mapa conseguimos obter algumas informações como o local onde fica, número de telefone, e o nome do banco. Este modo foi construído usando a API do GoogleMaps, à exceção do raio de visão que foi adicionado neste protótipo.

3.4 Tratamento de Sobreposições *On-Screen*

Um dos problemas recorrentes da RA é a facilidade com que ocorrem sobreposições de diferentes elementos virtuais, dificultando a visualização das informações fornecidas. Por isso, a aplicação desenvolvida preocupa-se em fazer o tratamento das sobreposições, permitindo

apresentar a informação de forma organizada e clara ao utilizador.

Optou-se, então, por transportar o conceito de pilha também para a área *on-screen*. De forma geral, a ideia é que sempre que dois pontos de interesse ocupem um espaço muito próximo, sejam agregados e apresentados como uma pilha, a qual, além de tomar a cor e transparência do mais relevante e mais próximo, assume também o seu tamanho (que indica a distância do ponto ao utilizador).

A pilha mostra um número indicativo da quantidade de pontos ali agregados. Ao contrário das pilhas formadas nas barras de sinalização, quando existe uma pilha na área visível é possível desagregá-la, clicando sobre a mesma. Quando uma pilha é desagregada, esta mostra separadamente os pontos que estavam agregados indicando a posição exata de cada um no ecrã. Os pontos desagregados voltam a agregar-se após 7 segundos. O resultado final da implementação do processo de tratamento de sobreposições é ilustrado na Figura 7.

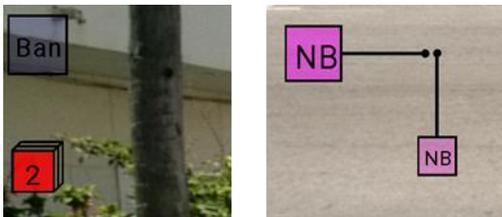


Figura 7: Tratamento de sobreposições:
a) agregação, e b) desagregação.

De seguida é explicado como é que o processo de agregação é feito internamente. Primeiramente, é necessário dividir o mundo em sectores circulares, para que a grelha criada para representar o mundo esteja fixa ao mesmo e não agarrada ao ecrã do dispositivo. Em seguida, para cada ponto, com base na sua localização geográfica, é calculado o azimute e a altitude, para determinar em que sector circular do mundo o ponto está situado.

A opção por trabalhar em coordenadas do mundo e não do ecrã tem a vantagem de evitar sucessivas agregações e desagregações de símbolos à medida que se muda a orientação do dispositivo. Ou seja, se a grelha estivesse associada às coordenadas de ecrã, ao rodar o dispositivo, a grelha rodaria. Assim a grelha passaria a estar associada a outra área do mundo e consequentemente as células corresponderiam também a outras sub-regiões. Isto implicaria que os pontos de interesse associados às células iriam variar e os símbolos de agregação também, havendo uma descontinuidade nas representações.

Dividimos o mundo horizontalmente e verticalmente em fatias de 9° cada e sabendo que a zona visível cobre uma área de 90° , o utilizador é capaz de ver 10 ($90^\circ/9$) fatias do mundo real *on-screen*. Os restantes 270° do mundo real representam as restantes 30 ($270^\circ/9$) fatias *off-screen*.

Como já foi referido, a grelha que divide o mundo em fatias depende das coordenadas do mundo (*lat, lon*) e não do ecrã, e, por essa razão, à medida que o utilizador muda

a sua orientação, a grelha não se move mas o utilizador passa a olhar para outras células.

Contudo, durante a maior parte do tempo, a orientação do utilizador não vai capturar na área visível exatamente 10 fatias mas sim 11, isto porque a câmara do dispositivo vai cobrir 9 fatias inteiras e outras 2 não totalmente cobertas, como ilustrado na Figura 8a.

Para resolver este caso, é verificada qual das 2 fatias tem maior proporção coberta e essa é a escolhida para ser representada inteiramente, descartando a outra. No caso da Figura 8a, como a fatia 1 é menos coberta do que a 11, é feito um ajuste à visualização, representando toda a última fatia e descartando a primeira como mostrado na Figura 8b. O mesmo raciocínio é aplicado às fatias verticais.

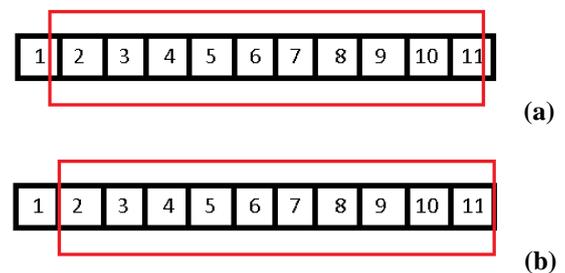


Figura 8: a) caso com fatias não totalmente cobertas, e b) resultado após o ajuste na visualização.

4. CONCLUSÕES E TRABALHO FUTURO

Neste artigo foi apresentado o protótipo de uma aplicação Android que permite localizar pontos de interesse em ambientes de realidade aumentada móvel.

Esta aplicação permite a navegação e exploração dos pontos de interesse em RA com sinalização de objetos *off-screen*, representação da relevância e tratamento de sobreposições, quer para pontos *on-screen*, quer para pontos *off-screen*. Além disto, fornece ainda um modo de Mapa 2D que poderá ser usado de forma complementar à RA, permitindo ao utilizador reconhecer mais facilmente a posição relativa de todos os objetos assinalados.

Estão em curso testes com utilizadores. Os resultados destes testes ajudarão a corrigir problemas que venham a ser detetados e a identificar as preferências dos utilizadores na combinação de RA com a visualização em mapas.

5. AGRADECIMENTOS

Agradecemos o apoio material da unidade de I&D BioISI, UID/MULTI/04046/2013, financiada pela FCT/MCTES/ PIDDAC e da unidade de I&D LaSIGE, através do projeto estratégico PEst-OE/EEI/UI0408/2014, Portugal.

6. REFERÊNCIAS

- [Aretz92] Aretz, A. J., & Wickens, C. D. (1992). The mental rotation of map displays. *Human performance*, 5(4), 303-328.
- [Azuma97] Azuma, R. T. (1997). A survey of augmented reality. *Presence*, 6(4), 355-385.

- [Boll14] Müller, H., Löcken, A., Heuten, W., & Boll, S. (2014). Sparkle: an ambient light display for dynamic off-screen points of interest. In *Proceedings of the ACM 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, pp. 51-60.
- [Burigat07] Burigat, S., & Chittaro, L. (2007). Navigation in 3D virtual environments: Effects of user experience and location-pointing navigation aids. *International Journal of Human-Computer Studies*, 65(11), 945-958.
- [Carlson-Radvansky93] Carlson-Radvansky, L. A., & Irwin, D. E. (1993). Frames of reference in vision and language: Where is above?. *Cognition*, 46(3), 223-244.
- [Carmo12] Carmo, M. B., Cláudio, A. P., Ferreira, A., Afonso, A. P., Simplício, R. (2012). Improving Symbol Saliency in Augmented Reality. In *Proceedings of the International Conference on Computer Graphics Theory and Applications, GRAPP 2013*, pp. 367-372.
- [Chitarro04] Chittaro, L., & Burigat, S. (2004). 3D location-pointing as a navigation aid in Virtual Environments. In *Proceedings of the ACM working conference on Advanced visual interfaces*, pp. 267-274.
- [Gonçalves11a] Gonçalves, T., Afonso, A. P., Carmo, M. B., & Pombinho, P. (2011). HaloDot: Visualization of the Relevance of Off-Screen Objects. In *Atas do V Simpósio Ibero-Americano de Computação Gráfica, SIACG 2011*, pp. 117-120.
- [Gonçalves11b] Gonçalves, T. (2011). *Visualização da relevância da informação geográfica em aplicações móveis*, Tese de mestrado em Engenharia Informática, DI-FCUL.
- [Jo11] Jo, H., Hwang, S., Park, H., & Ryu, J. H. (2011). Aroundplot: Focus+context interface for off-screen objects in 3D environments. *Computers & Graphics*, 35(4), 841-853.
- [Mulloni10] Mulloni, A., Dünser, A., & Schmalstieg, D. (2010). Zooming interfaces for augmented reality browsers. In *Proceedings of the ACM 12th international conference on Human computer interaction with mobile devices and services*, pp. 161-170.
- [Nóbrega15] Nóbrega, R., Cabral, D., Jacucci, G., & Coelho, A. (2015). NARI: Natural Augmented Reality Interface. In *Proceedings of the International Conference on Computer Graphics Theory and Applications, GRAPP*, pp. 504-510.
- [Pence10] Pence, H. E. (2010). Smartphones, smart objects, and augmented reality. *The Reference Librarian*, 52(1-2), 136-145.
- [Pombinho09] Pombinho, P., Carmo, M. B., & Afonso, A. P. (2009). Evaluation of overcluttering prevention techniques for mobile devices. In *Proceedings of the 13th International Conference on Information Visualisation*, pp. 127-134.
- [Schinke10] Schinke, T., Henze, N., & Boll, S. (2010). Visualization of off-screen objects in mobile augmented reality. In *Proceedings of the ACM 12th international conference on Human computer interaction with mobile devices and services*, pp. 313-316.
- [Siu13] Siu, T., & Herskovic, V. (2013). SidebARs: improving awareness of off-screen elements in mobile augmented reality. In *Proceedings of the ACM 2013 Chilean Conference on Human-Computer Interaction*, pp. 36-41.
- [Swienty08] Swienty, O., Reichenbacher, T., Reppermund, S., & Zihl, J. (2008). The role of relevance and cognition in attention-guiding geovisualisation. *The Cartographic Journal*, 45(3), 227-238.



Interactive Computer Graphics

Contents

Visualização de Fluxos Migratórios: Os Resultados das Candidaturas ao Ensino Superior Público <i>Tiago Brito, Joaquim Madeira, Beatriz Sousa Santos</i>	80
Modelação em Superfícies Interactivas para Fabricação 3D <i>Daniel Mendes, Pedro Pascoal, Miguel Nixo, Alfredo Ferreira</i>	88
Patient Specific Implants Design Tool for Surgeons <i>Gustavo Mações, Nuno Sousa, Nelson Alves, Nelson Fânzeres, Rui Coelho</i>	96
Mid-Air Manipulation of 3D Models in (Semi-)Immersive Virtual Environments <i>Vasco Rodrigues, Daniel Mendes, Alfredo Ferreira, Joaquim Jorge</i>	100

Visualização de Fluxos Migratórios: Os Resultados das Candidaturas ao Ensino Superior Público

Tiago Brito
DETI
Universidade de Aveiro
Aveiro
tlfbrito@ua.pt

Joaquim Madeira
DETI / IEETA
Universidade de Aveiro
Aveiro
jmadeira@ua.pt

Beatriz Sousa Santos
DETI / IEETA
Universidade de Aveiro
Aveiro
bss@ua.pt

Resumo

Os candidatos ao Ensino Superior Público são colocados nas diferentes instituições através do Concurso Nacional de Acesso (CNA), o que pode originar a sua migração para outro distrito ou região autónoma. São propostas três visualizações interativas — representação matricial, diagrama cordal e representação georreferenciada — que permitem a apresentação, exploração e análise dos resultados do CNA, sendo usados os dados relativos aos candidatos colocados na 1ª fase dos concursos de 2012, 2013 e 2014. Estas visualizações foram avaliadas por três processos: avaliação heurística, testes de usabilidade supervisionados e recorrendo à utilização do Amazon Mechanical Turk (AMT).

Palavras-Chave

Visualização, fluxos migratórios.

1. INTRODUÇÃO

Os jovens que concluem o Ensino Secundário e se candidatam ao Ensino Superior Público português são colocados, nas diferentes instituições de ensino universitário ou politécnico, através do Concurso Nacional de Acesso (CNA). Embora muitos dos candidatos venham a frequentar instituições do seu distrito ou região autónoma (DRA) de origem, outros são colocados em cursos e instituições que obrigam à sua *migração* no interior do território nacional.

No âmbito da dissertação de mestrado do primeiro autor [Brito 15], foram desenvolvidas e avaliadas três visualizações interativas — representação matricial, diagrama cordal e representação georreferenciada — que permitem apresentar, explorar e analisar os fluxos migratórios originados pelas colocações do CNA. Essas visualizações, e os dados relativos aos cerca de 115.000 candidatos colocados na 1ª fase dos concursos nacionais de 2012, 2013 e 2014, estão disponíveis numa plataforma *Web*.¹

Nas duas próximas secções são referidos alguns trabalhos na área de Visualização de Fluxos Migratórios e descritos os dados do CNA que foram disponibilizados pela Direção-Geral do Ensino Superior (DGES). Na secção seguinte são apresentadas as três visualizações propostas, e as suas características mais importantes, bem como a plataforma criada para as disponibilizar ao público. De seguida são descritos os processos de avaliação da plataforma e das

visualizações desenvolvidas — avaliação heurística, testes de usabilidade supervisionados e recorrendo à utilização do *Amazon Mechanical Turk (AMT)* —, e apresentados alguns dos seus resultados. Na última secção são apresentadas algumas conclusões e ideias para trabalho futuro, bem como exemplificados alguns resultados da exploração dos resultados do CNA usando as visualizações desenvolvidas.

2. VISUALIZAÇÃO DE FLUXOS MIGRATÓRIOS

A visualização de fluxos migratórios é uma das áreas de aplicação da Visualização de Dados e de Informação que, tal como outras, tem as suas raízes em representações surgidas no século XIX.

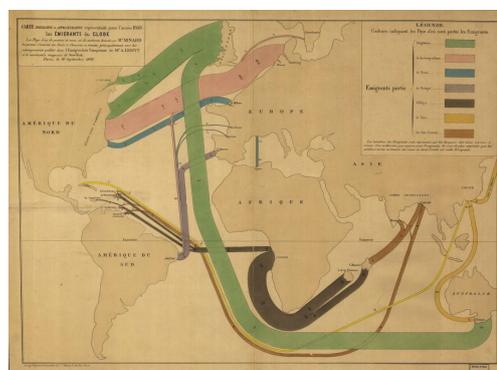


Figura 1. Fluxos de emigração em 1858.²

¹www.portugal-migration.info

²C. Minard, 1862, www.datavis.ca/gallery/minbib.php

Uma dessas primeiras visualizações parece ser o mapa-mundo de Charles Minard apresentando os fluxos de emigração para o ano de 1858, em que a espessura das faixas representando os fluxos é proporcional ao número de emigrantes e é usada cor para melhor identificar a origem de cada fluxo migratório (ver Fig. 1).

As primeiras representações de fluxos migratórios sobre mapas, desenvolvidas com recurso a computadores, aparecem nas décadas de 1960 e 1970, surgindo mais tarde as primeiras ferramentas de software de suporte à sua criação. Uma dessas ferramentas mais referenciadas é o *Flow Mapper*, cuja primeira versão foi desenvolvida por W. Tobler na década de 1980.

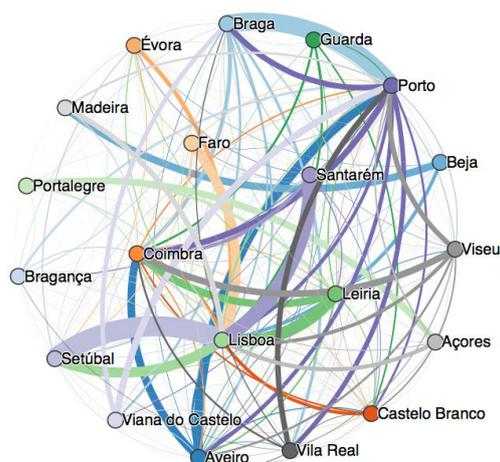


Figura 2. Representação ingênua, usando um grafo, dos candidatos colocados no Ensino Superior em 2012.

A representação de fluxos migratórios usando grafos orientados é uma alternativa simples e intuitiva à representação georreferenciada: os nós do grafo representam a origem e o destino de cada fluxo; os atributos de cada fluxo são, por exemplo, mapeados na espessura e na cor do correspondente arco. No entanto, se o grafo for (quase) completo, e mesmo que o número de nós seja pequeno, este tipo de representação direta torna-se inútil (ver Fig. 2), a menos que se use alguma estratégia para posicionamento dos nós e agrupamento de arcos.

Atendendo a este problema, John Bryant propôs a visualização matricial da Figura 3: o grafo orientado é representado pela sua matriz de adjacências e o tamanho do quadrado associado a cada elemento da matriz é proporcional ao valor do correspondente fluxo migratório [Bryant 11]. Trata-se de um tipo de representação familiar, por semelhança com as representações em tabela, que permite representar um grande número de fluxos sem sobreposições, mas pode não ser fácil avaliar e comparar o valor dos vários fluxos migratórios.

Fluxos de diferentes tipos são habitualmente representados por Diagramas de Sankey. No caso dos fluxos migratórios é usado um diagrama vertical simples: as origens estão re-

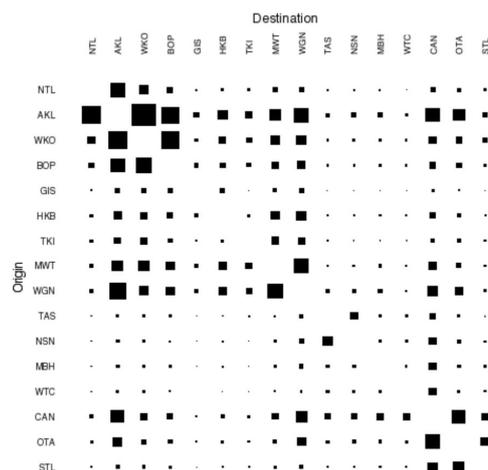


Figura 3. Visualização das migrações internas na Nova Zelândia [Bryant 11].

presentadas do lado esquerdo e os destinos do lado direito; a espessura dos arcos unindo origens e destinos é proporcional ao número de migrantes. No entanto, estes diagramas verticais podem apresentar problemas de usabilidade devido ao seu tamanho: o diagrama pode não ser totalmente representável no ecrã e ser difícil seguir a maioria dos fluxos, mesmo efetuando deslocamentos verticais. Já depois do início do trabalho descrito nesta comunicação, a DGES inaugurou um *site* que apresenta estatísticas relativas aos candidatos ao Ensino Superior em 2014, disponibilizando o Diagrama de Sankey da Figura 4.

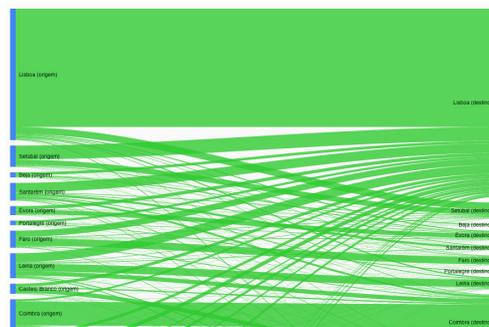


Figura 4. Vista parcial do Diagrama de Sankey representando as colocações dos candidatos ao Ensino Superior em 2014.⁴

A representação em diagrama cordal apresenta um *layout* alternativo às visualizações anteriores: as origens e destinos dos fluxos são dispostos radialmente ao longo de uma circunferência; os fluxos entre origens e destinos são representados por cordas (i.e., arcos) de diferentes cores e de espessura proporcional ao número de migrantes — algumas destas visualizações uma única corda de espessura variável representa, para cada par (origem, destino), os flu-

⁴DGES, www.dges.mec.pt/estatisticasacesso/

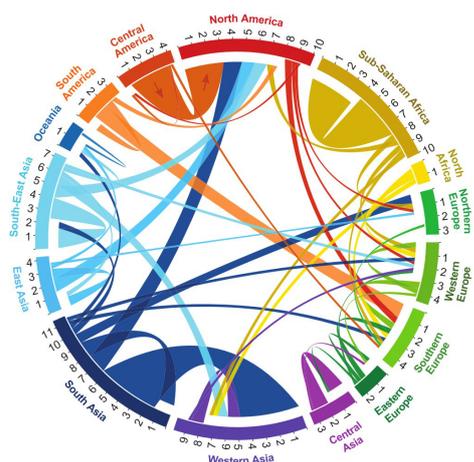


Figura 5. Diagrama cordal representando os fluxos migratórios entre várias regiões do globo [Abel 14].

xos migratórios em ambos os sentidos. O grupo de N. Sander vem desenvolvendo visualizações intuitivas de fluxos migratórios com base em diagramas cordais (ver Fig. 5) [Abel 14].

3. DADOS DO CONCURSO NACIONAL DE ACESSO

O Concurso Nacional de Acesso (CNA) ao Ensino Superior Público é, em termos do número de candidatos, visibilidade e impacto social, um dos mais importantes concursos anuais realizados pela Administração Pública. Para os últimos três anos, o número de candidatos e de colocados na primeira fase do CNA é apresentado na tabela seguinte (Tab. 1).

Ano	Candidatos	Colocados	%
2012	45.078	40.415	90%
2013	40.419	37.415	93%
2014	42.408	37.778	89%

Tabela 1. Concurso Nacional de Acesso: número de candidatos e de colocados na 1ª fase (2012 a 2014).

A DGES disponibilizou aos autores dados relativos aos cerca de 115.000 candidatos colocados na 1ª fase dos concursos nacionais de 2012, 2013 e 2014. Esses dados foram fornecidos num ficheiro Excel com 6 folhas de trabalho: 2012, 2013, 2014, GAES⁵, Estabelecimento e Distrito.

Nas três primeiras folhas, e para o respetivo ano, estão listados os atributos de cada candidato colocado (ver Tab. 2); as restantes folhas contêm informação complementar relativas às colunas (i.e., atributos) codificados. A folha GAES associa ao código de cada GAES a sua localização, que é usada para identificar o distrito ou região autónoma de ori-

⁵Gabinete de Acesso ao Ensino Superior.

Atributo	Descrição
Ano	Ano de candidatura
IDAluno	Identificador único
FaseCand	Fase de candidatura (apenas 1ª fase)
GAES	Código do GAES de origem
EstabelCol	Código da instituição de colocação
NotaCand	Nota de candidatura
AnoNasc	Ano de nascimento
Sexo	Género

Tabela 2. Atributos associados aos candidatos colocados.

gem dos colocados. A folha Estabelecimento permite associar ao código de cada estabelecimento de ensino a correspondente instituição em que cada candidato é colocado (universidade, faculdade, instituto politécnico, etc.).

Os dados recebidos da DGES foram processados para (i) verificar a sua integridade, (ii) remover informação considerada desnecessária (p.ex., a indicação da fase de colocação), (iii) agregar-lhes a população de cada distrito ou região autónoma, e (iv) optimizá-los para leitura, exportando-os para ficheiros CSV.

Para facilitar a análise dos dados foram definidos os seguintes indicadores numéricos, em que $c(i, j)$ representa o número de candidatos oriundos de i e colocados em j , sendo i e j distritos ou regiões autónomas:

- Total de candidatos colocados em d :
 $Tcol(d) = \sum_i c(i, d)$
- Total de candidatos oriundos de d :
 $Tori(d) = \sum_j c(d, j)$
- Saldo de colocados para d :
 $S(d) = Tcol(d) - Tori(d)$
- Rácio de colocados para d :
 $R(d) = [Tcol(d)/Tori(d)] \times 100$
- Saldo de colocados entre u e v :
 $S(u, v) = c(u, v) - c(v, u)$
- Percentagem de candidatos *migrando* de u para v :
 $M(u, v) = [c(u, v)/Tori(u)] \times 100$

Os valores destes indicadores são usados em legendas e tabelas auxiliares, ou mapeados em atributos visuais (cor, espessura e/ou tamanho) nas três visualizações propostas.

4. VISUALIZAÇÕES DESENVOLVIDAS

A colocação dos candidatos ao Ensino Superior nas diferentes instituições pode ser encarada como uma *migração*. Assim, podem ser desenvolvidas/adaptadas técnicas de representação de fluxos migratórios para criar visualizações interativas que possibilitam a representação, exploração e análise dos resultados anuais do CNA.

Descrevem-se de seguida as três visualizações desenvolvidas — representação matricial, diagrama cordal e

representação georreferenciada —; a biblioteca D3.js [Bostock 11] foi usada como base da implementação.

4.1. Representação Matricial

Esta visualização (ver Fig. 6) tem por base a matriz de adjacências do grafo orientado associado à colocação dos candidatos, de modo semelhante ao trabalho de J. Bryant. Há, no entanto, diferenças significativas que serão explicadas de seguida.

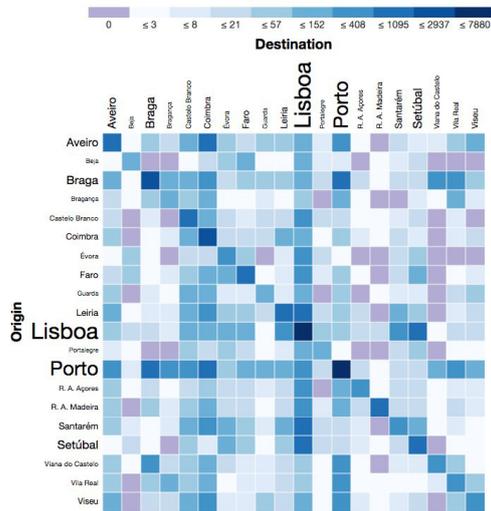


Figura 6. Visualização matricial, representando o número de candidatos colocados, com a correspondente escala de cores.

Como se pode ver na Figura 6, as linhas e colunas da matriz correspondem aos distritos e regiões autónomas (DRA) de origem dos candidatos e em que os candidatos são colocados. Para destacar os distritos mais populosos, o tamanho das letras da correspondente designação é proporcional à população de cada DRA. Acima da matriz é colocada a escala de cores correspondente aos dados representados.

Para cada conjunto de dados selecionado, e para cada par (origem, destino) — i.e., para cada fluxo de candidatos colocados —, é possível visualizar dois conjuntos de valores: $c(i, j)$, o número de colocados, ou $S(i, j)$, o saldo das colocações.

Para evitar enganos, são usadas duas escalas de cores: o número de candidatos colocados é mapeado de modo logarítmico numa escala de cores com 9 tonalidades de azul; o saldo positivo, nulo ou negativo de colocados é representado pelas cores verde, cinzento e encarnado, respetivamente. No caso de não existirem candidatos colocados, a correspondente célula da matriz é preenchida a roxo, ficando claramente destacada das restantes células. A escolha do conjunto de valores a representar desencadeia um conjunto de transições animadas de forma a adaptar progressivamente a visualização à nova escala de cores.

Uma das características particulares desta visualização é a capacidade de ordenar os distritos e regiões autónomas (i.e., as linhas e colunas da matriz) segundo quatro

critérios: designação, número total de candidatos, número total de colocados ou população. Este tipo funcionalidade é especialmente útil para comparar diferenças e destacar padrões, tal como sugerido por [Haroz 12].

4.2. Diagrama Cordal

Esta visualização (ver Fig. 7) é a única desenvolvida com base num *layout* D3.js predefinido (*Chord Layout*). No entanto, esta é uma das representações mais difíceis de gerar, pois necessita de estruturas de dados particulares para armazenar os dados e os outros elementos adicionais (texto, cores, etc.).

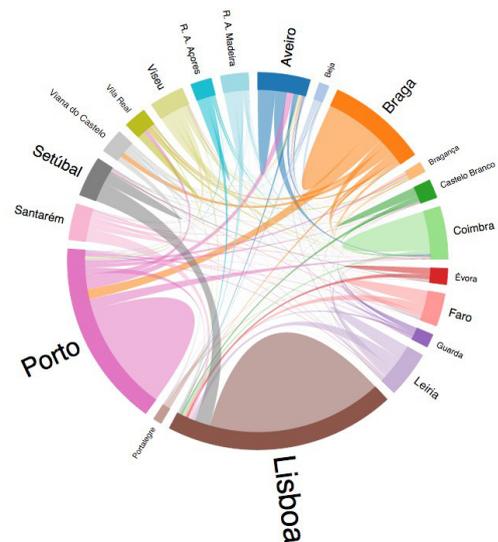


Figura 7. Visualização usando o diagrama cordal.

Tal como é habitual nos diagramas cordais, na periferia estão representados os vários DRA e estes encontram-se repartidos de acordo com a proporção dos candidatos colocados de cada DRA. Tal como na representação matricial, o tamanho da designação de cada DRA é proporcional à respetiva população.

Embora seja uma representação semelhante à referida na Secção 2, foram tomadas algumas opções particulares:

- Cada fluxo de colocados é representado, a menos que seja nulo, não se estabelecendo um (habitual) limite mínimo de colocados para a sua representação.
- A cada DRA é associada uma cor diferente, escolhida de um conjunto de cores de 20 tons divergentes.
- Entre cada par (origem, destino) é apenas desenhado um arco (i.e., uma corda) de espessura variável, que representa os dois fluxos com sentidos opostos, o que diminui o número de arcos representados.
- A espessura em cada extremidade de um arco é proporcional ao número de candidatos colocados no correspondente DRA.

- A cor de cada arco é a cor da sua extremidade (i.e., DRA) que originou maior número de candidatos colocados.
- O arco que representa o conjunto de candidatos de um DRA colocados nesse mesmo DRA é representado como uma *bossa*, com a cor do DRA associado.
- Os arcos associados a cada DRA estão ordenados, em sentido horário, pelo número dos correspondentes candidatos colocados.⁶

A seleção, por filtragem, de um novo conjunto de dados a ser representado origina uma interpolação do estado atual para o estado final, sendo, por exemplo, removidos os arcos desnecessários, adicionados eventuais novos arcos e reposicionados todos os componentes do diagrama de acordo com os dados recém-selecionados. Assim, surge um efeito de sucessão do estado anterior para o subsequente, que não existe numa implementação mais tradicional em que o diagrama é subitamente redesenhado, e o utilizador pode mais facilmente perceber diferenças entre os dois estados.

4.3. Representação Georreferenciada

A representação georreferenciada desenvolvida (ver Fig. 8) permite visualizar os fluxos de candidatos colocados sobre o mapa de Portugal e das suas duas Regiões Autónomas.

Nesta visualização é associada (por omissão) uma cor a cada distrito ou região autónoma (DRA), usando 20 tons divergentes, permitindo uma clara associação entre cada cor e um DRA, o que ajuda o utilizador a identificar cada distrito e a sua fronteira. É possível alterar este mapeamento para uma escala de tons de azul representando a população de cada DRA: quanto mais escuro for o tom, maior o número de habitantes.

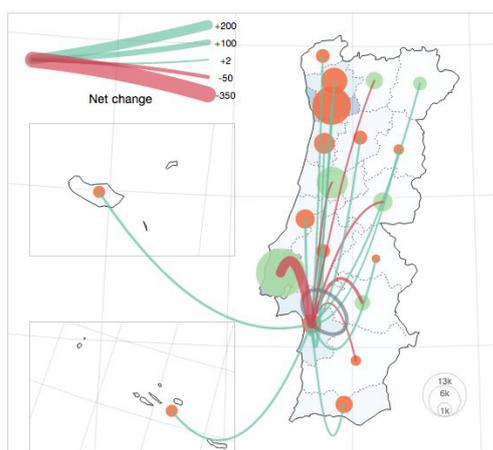


Figura 8. Visualização georreferenciada.

A cada DRA está associado um círculo de raio variável, posicionado no seu centróide, que permite a fácil seleção

⁶Assim, é fácil identificar os casos em que o maior número de candidatos é colocado ou não no mesmo DRA.

do respetivo DRA: pormenor importante no caso das regiões autónomas e de distritos de menor área; evita-se também mapear qualquer atributo usando a forma/área de cada DRA. O círculo selecionado é realçado por uma linha negra ao longo do seu perímetro, para facilitar a perceção da sua seleção, uma vez que os fluxos de candidatos colocados serão representados com origem no círculo selecionado.

O raio de cada um destes círculos é proporcional ao valor do indicador numérico selecionado para a visualização: número total de colocados ($Tcol(d)$), número total de candidatos ($Tori(d)$) ou rácio de colocados ($R(d)$). Já a cor de cada círculo é definida pelo valor de outro indicador, o saldo de colocados ($S(d)$): encarnado para um saldo negativo, verde para um saldo positivo e cinzento para um saldo nulo.

Os arcos que representam os fluxos de candidatos colocados são desenhados entre os círculos associados aos DRA, sendo a cor e espessura do arco (u, v), e com destino em v , definidas do seguinte modo:

- A espessura do arco é proporcional ao valor absoluto do saldo de colocados, $S(u, v)$.
- A cor do arco será verde-claro se $S(u, v) > 0$, encarnado se $S(u, v) < 0$, ou cinzento se $S(u, v) = 0$.
- O arco com origem e destino em u (lacete), representando os candidatos oriundos de u e aí colocados, é processado de modo particular, sendo-lhe atribuída a cor cinzento.

Enquanto que cada lacete é desenhado como um arco de elipse, os outros arcos são desenhados como curvas quadráticas de Bézier, usando a linguagem SVG.

O modo como o valor do saldo de candidatos colocados é mapeado nos vários arcos permite distinguir, de forma clara, as situações em que esse saldo é positivo ou negativo, bem como ter uma ideia do seu valor. Um efeito interessante é conseguido desenhado os arcos com uma animação simples, após a seleção de um DRA: os arcos associados a saldos positivos são desenhados com destino no DRA selecionado, enquanto que os arcos associados a saldos negativos são desenhados com origem no DRA selecionado.

Devido à diversidade de elementos visuais e variáveis mapeadas em simultâneo, esta representação contém duas legendas de forma a auxiliar o utilizador na interpretação da informação visual existente.

4.4. Plataforma Web

As visualizações desenvolvidas foram alojadas numa plataforma *Web* (www.portugal-migration.info), tendo como objetivos principais facilitar o acesso às visualizações, contextualizar o seu âmbito e descrever as várias funcionalidades num mesmo local, auxiliando os utilizadores na compreensão e exploração dos dados disponibilizados. No sentido de internacionalizar a utilização da plataforma, foi escolhida a língua inglesa para a sua

apresentação. A página de entrada contém uma introdução a cada uma das visualizações, acompanhada de uma imagem ilustrativa, visível na Figura 9.

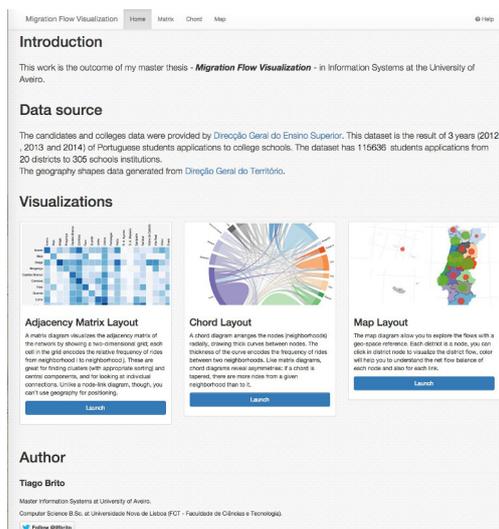


Figura 9. Página de entrada da plataforma desenvolvida.

Cada uma das visualizações é acessada através do seu botão de ação e acompanhada por um painel de filtros (comuns às três visualizações) e controlos que permitem filtrar a informação existente (por género dos candidatos, ano de candidatura e tipo de instituição) e interagir com cada visualização.

Estão também disponíveis de forma transversal às três visualizações: uma *tooltip* que apresenta os indicadores numéricos respetivos quando é selecionado um distrito ou um fluxo de colocados; uma tabela que apresenta a lista de instituições e o número de colocados em cada uma delas, para um dado fluxo de colocados selecionado.

5. AVALIAÇÃO

As características e funcionalidades da plataforma *Web* e das visualizações interativas desenvolvidas foram avaliadas através de métodos de avaliação heurística, testes de usabilidade supervisionados e recorrendo à utilização do *Amazon Mechanical Turk (AMT)*, para detetar eventuais limitações e avaliar a sua eficácia.

5.1. Avaliação Heurística

Esta avaliação foi efetuada de modo independente por dois avaliadores⁷, com base nos modelos de Forsell, de Nielsen e de Zuk [Forsell 10, Nielsen 95, Zuk 06], de modo a avaliar a gravidade dos problemas de usabilidade detetados através de duas métricas: aplicabilidade da heurística (escala de 0 a 5) e gravidade do problema (escala de 0 a 4).

Os problemas mais importantes detetados, e as soluções encontradas, foram os seguintes:

⁷Alunos do curso de Mestrado em Sistemas de Informação da U. Aveiro, sem relação com o primeiro autor.

- As escalas de cores originais não eram corretamente distinguíveis por daltónicos, originando interpretações enviesadas e perda de informação — os tons das escalas de cores foram alterados para minorar essa perda de informação.
- O estado dos elementos do painel de filtros e controlos não era apresentado de modo uniforme, nomeadamente para a seleção do género dos colocados — foi feita uma correção para garantir a uniformização desejada.
- Ajuda de contexto insuficiente, dado que, além de ser usada a língua inglesa, ocorre um conjunto de termos que não são intuitivos para o público em geral — foi introduzido um mecanismo de ajuda contextual guiada, transversal a todas as visualizações, que apresenta uma descrição de todos os filtros e controlos e destaca a funcionalidade em contexto.

O resultado global das duas avaliações heurísticas foi considerado positivo e construtivo, confirmando a consistência existente entre as características das três visualizações, e o modo como são apresentadas e podem ser usadas na plataforma. Foi também destacada a intenção subentendida a cada visualização, de que cada uma delas melhor se adequa a realizar algumas tarefas específicas.

5.2. Teste de Usabilidade

O teste de usabilidade compreendeu a realização, por um conjunto de 12 utilizadores, de uma sequência de tarefas com propósito bem definido, e seguiu a metodologia habitual: numa primeira fase foi efetuada uma apresentação da plataforma e das suas funcionalidades, e dada uma explicação dos procedimentos do teste; de seguida, os utilizadores realizaram as tarefas predefinidas e avaliaram o grau de dificuldade de cada uma, usando uma escala de Likert (5 níveis). A prestação de cada utilizador foi acompanhada por um observador que, entre outros pormenores, registou o tempo necessário à conclusão de cada tarefa e o modo como foi executada.

As tarefas propostas foram divididas em dois grupos:

1. Para cada uma das visualizações, foram solicitadas tarefas distintas, mas equivalentes na sua dificuldade e no seu objetivo, de modo a introduzir e ambientar os utilizadores às visualizações e exibir as suas diferenças principais.
2. Sem que fosse indicado o tipo de visualização a usar, foi novamente solicitada a realização de tarefas, devendo o utilizador escolher a visualização que julgasse mais apropriada. A resolução de algumas tarefas era mais fácil através de uma escolha adequada, ou mesmo impossível sem a escolha da única visualização apropriada.

Cada um dos 12 utilizadores realizou um total de 21 tarefas: o primeiro grupo continha 5 tarefas para cada visualização e segundo grupo 6 tarefas adicionais.

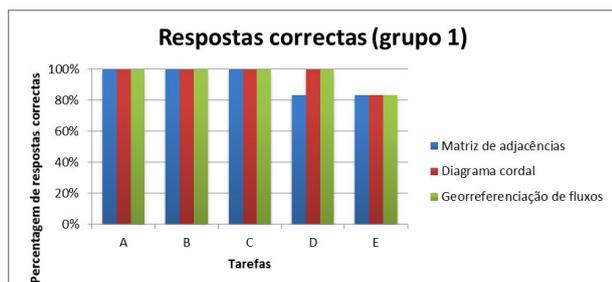


Figura 10. Percentagem de tarefas concluídas corretamente para o primeiro grupo.

Para o grupo 1, os resultados mostram que os participantes conseguiram concluir todas as tarefas; no entanto, a tarefa E⁸ foi errada por 20% dos participantes, para todas as visualizações (ver Fig. 10), o que pode indicar um eventual problema de interpretação do seu enunciado.

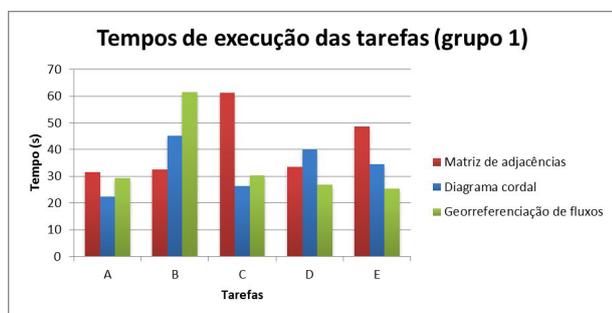


Figura 11. Tempo médio de conclusão das tarefas do primeiro grupo.

O gráfico da Figura 11 mostra que, apesar de todos os participantes terem concluído as tarefas do primeiro grupo, a sua resolução foi realizada com tempos bastante distintos. Destacam-se, sobretudo, os tempos médios mais longos para a tarefa B, utilizando a representação georreferenciada, e para a tarefa C, usando a representação matricial.

Para as tarefas do segundo grupo, o gráfico da Figura 12 mostra uma categorização relativamente consistente quanto à facilidade de execução das tarefas propostas. Fica também claro que grande parte dos participantes optou por utilizar a representação matricial, exceto nas tarefas D e F, onde a resolução era apenas possível através da representação georreferenciada. Isto indica que, apesar dos participantes preferirem utilizar a representação matricial, conseguem identificar claramente a visualização mais correta para resolver tarefas de características georreferenciadas.

As críticas e os comentários apontados durante o teste de usabilidade são também uma contribuição valiosa dos participantes. Destacam-se os julgados mais importantes:

⁸Identificar o distrito que recebe o maior número de colocados oriundos de um dado distrito X, excluindo X.

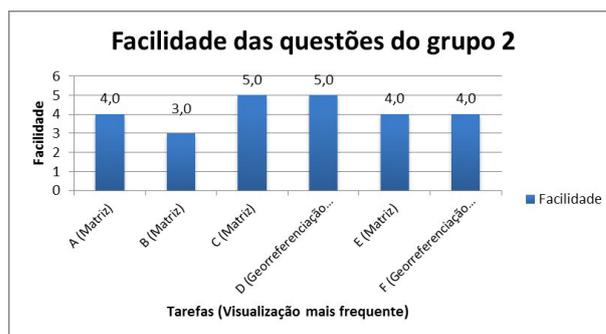


Figura 12. Mediana do nível de facilidade atribuído e visualização mais frequente escolhida, para as tarefas do segundo grupo.

- Dificuldade em encontrar os totais dos DRA na representação matricial.
- Dificuldade na análise das visualizações cordal e georreferenciada, devido ao número de arcos presentes.
- Na visualização georreferenciada, os lacetes não têm o destaque necessário e são difíceis de descobrir.

5.3. Amazon Mechanical Turk

Apesar dos testes de usabilidade supervisionados serem proveitosos em termos dos dados e comentários recolhidos, a sua preparação e realização são trabalhosas, sendo difícil angariar participantes. Por este motivo, nos últimos anos tem assumido importância a avaliação das características de visualizações usando *crowdsourcing*, em que os participantes nas avaliações são recrutados através de serviços especializados, como o *Amazon Mechanical Turk (AMT)* [Kittur 08].

Após uma análise exploratória do *AMT*, foi decidido efetuar uma avaliação das visualizações desenvolvidas usando essa ferramenta. Para isso foram desenvolvidos três guiões — um para cada visualização a avaliar —, equivalentes nos seus objetivos e na dificuldade das suas questões/tarefas. Cada guião tinha 10 questões: 3 questões demográficas, 1 teste de daltonismo, 2 questões de validação do perfil do participante e 4 tarefas específicas para a visualização a avaliar. O processo de avaliação usando o *AMT* decorreu num período de 24 horas, com 90 participantes (30 para cada guião), com origem em diferentes países.

Entre outros resultados, foram registados o tempo de conclusão de cada uma das tarefas e o número de respostas corretas de cada participante. Uma análise preliminar revelou que os valores do tempo médio de conclusão de todas as tarefas são próximos quando são usados a representação matricial e o diagrama cordal (237 e 220 segundos), com um valor médio mais elevado quando é usada a representação georreferenciada (264 segundos). Esta última representação também originou um menor número médio de respostas corretas (64%), número que é mais elevado para a utilização do diagrama cordal (76%) e

da representação matricial (84%). A menor familiaridade com a geografia de Portugal, por parte dos participantes, talvez possa explicar, em parte, estes resultados.

6. CONCLUSÃO

Foram desenvolvidas, avaliadas e disponibilizadas na *Web* três visualizações interativas para a representação, exploração e análise dos fluxos de colocação dos candidatos ao Ensino Superior Público, a partir dos distritos e regiões autónomas de origem (DRA) e sendo usados os dados (2012 a 2014) da 1ª fase do Concurso Nacional de Acesso.

De notar que se trata de um tema com interesse para o público em geral, e que tem associado um conjunto de dados de tamanho expressivo, com um número razoável de atributos associados a cada candidato colocado

Apresentam-se de seguida alguns resultados globais da exploração dos dados, que exemplificam a utilidade das visualizações desenvolvidas:

- O mais importante fluxo de colocados oriundos de um DRA corresponde sempre aos colocados nesse DRA, exceto para o distrito de Setúbal.
- Para os candidatos oriundos dos distritos de Lisboa, Porto, Castelo Branco e Coimbra, a percentagem de colocados no seu próprio distrito é superior a 50%.
- Cada um dos outros DRA coloca mais candidatos no distrito de Lisboa do que o inverso.
- Setúbal é o único DRA de onde o mais importante fluxo de colocados não só não é para o próprio distrito, como também este é superior a 50% (para o distrito de Lisboa).
- Évora e Beja são os distritos cujos os candidatos são colocados num menor número de DRA — Évora: 7 distritos de colocação em 2012 e 2013; Beja: 8 distritos de colocação em 2014.
- A Região Autónoma da Madeira (2013 e 2014) e o distrito de Viana do Castelo (2012 a 2014) são os DRA que recebem colocados de um menor conjunto de DRA: um total de 10 DRA; de outros 10 não recebem qualquer colocado.

Como trabalho futuro referem-se: a vantagem em implementar mais sugestões resultantes da avaliação heurística e do teste de usabilidade supervisionado; o desenvolvimento de funcionalidades adicionais com base noutros atributos de cada colocado, como a nota de candidatura; e o interesse em manter a plataforma atualizada, acrescentando anualmente os dados das colocações, que irão sendo solicitados à DGES. Seria também muito interessante obter atributos adicionais associados a cada candidato colocado e dados adicionais de cada instituição. Por exemplo, permitindo analisar os diversos cursos em que os candidatos são colocados, o número de vagas existentes e sobrantes, etc.

Terá também de ser considerada a aplicação das visualizações desenvolvidas a outro tipo de cenários, no sentido de avaliar a sua eficácia e escalabilidade, permitindo uma visualização hierárquica e/ou seletiva dos dados, quando o número de pares (origem, destino) aumentar significativamente.

7. AGRADECIMENTOS

Agradece-se à Direção-Geral do Ensino Superior a disponibilização dos dados utilizados neste trabalho.

Agradece-se, também, a todos os que colaboraram na avaliação das visualizações desenvolvidas quer realizando as avaliações heurísticas quer participando nos testes de usabilidade.

Referências

- [Abel 14] G. J. Abel e N. Sander. Quantifying global international migration flows. *Science*, 343(6178):1520–1522, 2014.
- [Bostock 11] M. Bostock, V. Ogievetsky, e J. Heer. D3 data-driven documents. *IEEE Trans. Vis. Comp. Graphics*, 17(12):2301–2309, Dec 2011.
- [Brito 15] T. Brito. Visualização de fluxos migratórios: Os candidatos ao ensino superior público (2012 a 2014). Dissertação de Mestrado em Sistemas de Informação, Universidade de Aveiro, 2015.
- [Bryant 11] J. Bryant. Visualising internal migration flows. *New Zealand Population Review*, 37:159–171, 2011.
- [Forsell 10] C. Forsell e J. Johansson. An heuristic set for evaluation in information visualization. Em *Proc. Int. Conf. Advanced Visual Interfaces, AVI '10*, páginas 199–206. ACM, 2010.
- [Haroz 12] S. Haroz e D. Whitney. How capacity limits of attention influence information visualization effectiveness. *IEEE Trans. Vis. Comp. Graphics*, 18(12):2402–2410, Dec 2012.
- [Kittur 08] A. Kittur, E. H. Chi, e B. Suh. Crowdsourcing user studies with mechanical turk. Em *Proc. SIGCHI Conf. Human Factors in Computing Systems, CHI '08*, páginas 453–456. ACM, 2008.
- [Nielsen 95] J. Nielsen. How to conduct a heuristic evaluation, 1995.
- [Zuk 06] T. Zuk, L. Schlesier, P. Neumann, M. S. Hancock, e S. Carpendale. Heuristics for information visualization evaluation. Em *Proc. AVI Workshop BEyond Time and Errors: Novel Evaluation Methods for Information Visualization, BELIV '06*, páginas 1–6. ACM, 2006.

Modelação em Superfícies Interactivas para Fabricação 3D

Daniel Mendes Pedro Pascoal Miguel Nixo Alfredo Ferreira
 INESC-ID / IST / University of Lisbon

{danielmendes, pmbp, miguelnixo, alfredo.ferreira}@tecnico.ulisboa.pt

Resumo

A disponibilidade e utilização de objectos 3D aumentou durante os últimos anos, a par com a redução do custo da impressão tridimensional de objectos. Tal cenário torna necessário que existam ferramentas para a criação e modificação destes objectos de forma eficiente e natural, permitindo que estas possam ser usadas tanto por utilizadores experientes como inexperientes. Com este trabalho, foi nosso objectivo desenvolver uma solução para modelação 3D com foco em posterior impressão, que retira partido de superfícies interactivas, bem como de métodos de recuperação de objectos 3D. De forma a permitir uma experiência de interacção o mais natural possível, criámos e discutimos abordagens que são baseadas apenas em toque, caneta, ou numa combinação de ambas. Depois de uma fase de avaliação com utilizadores, os resultados sugerem que a abordagem baseada em interacção exclusivamente com caneta, apesar de satisfatória e agradável, não é tão eficiente quanto as restantes.

Palavras-Chave

Modelação, Impressão 3D, Superfícies Interactivas, Multi-Toque, Caneta, Recuperação de Objectos

1. INTRODUÇÃO

A tecnologia de impressão 3D e respectiva comunidade têm tido crescimentos significativos nos últimos anos, com o preço médio da impressora 3D padrão a diminuir substancialmente durante esse período. Tendo isto em conta, é previsível que a impressão 3D será tornada gradualmente mais disponível para o utilizador comum, tornando possível fabricar modelos 3D impressos em casa com custos relativamente baixos. Simultaneamente, a cada vez mais elevada disponibilidade de objectos 3D na Internet reforça a importância do desenvolvimento de soluções de modelação 3D que permitam a criação de novos modelos ou a modificação de existentes com facilidade. Embora muito já tenha sido feito para desenvolver ferramentas profissionais, estas requerem conhecimentos avançados e formação de modo a poderem ser utilizadas de forma eficaz, mesmo que o objetivo seja criar uma forma simples.

Além disso, superfícies interactivas com multi-toque, cuja tecnologia já existe há vários anos, estão-se a tornar cada vez mais desejadas e populares, estando presentes em dispositivos do quotidiano. Estas superfícies são amplamente utilizadas em *smartphones*, *tablets*, computadores portáteis e mesmo mesas e quadros interactivos. Desta forma, representam uma alternativa viável à interacção WIMP (baseada em janelas, ícones, menus e apontador). O aparecimento de novas tecnologias e dispositivos originam naturalmente novas possibilidades para o desenvolvimento de novas técnicas de interacção, as quais deverão ser mais familiares para o utilizador comum. Para isso, analogias naturais podem ser utilizadas, tal como a manipulação bi-manual de objectos ou o uso de uma caneta para desenhar.

Embora o uso de superfícies sensíveis ao toque e interfaces caligráficas tenha sido estudado em diversos trabalhos, várias questões continuam por responder, nomeadamente sobre as vantagens e desvantagens de cada um e sobre o seu uso combinado em cenários de modelação 3D.

Pretendeu-se, com este trabalho, desenvolver uma abordagem de modelação 3D focada na fabricação 3D, que tire proveito de superfícies interactivas e de métodos para a recuperação de objetos 3D. O desenvolvimento de uma interface de utilizador natural (NUI - *natural user interface*) deverá permitir que o utilizador comum possa ser capaz de utilizar a solução de uma forma adequada e natural, conseguindo criar e imprimir um objeto idealizado e modelado por si, como exemplificado na Figura 1. Tomamos como hipótese que o uso combinado de toque e caneta permitirá que a modelação 3D possa ser mais atraente e acessível a utilizadores noviços, quando comparado a abordagens que utilizam um único ponto de contacto ou interfaces exclusivamente tácteis.

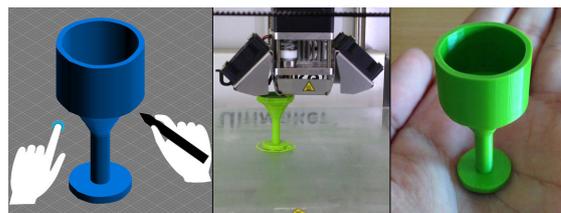


Figura 1. Exemplo de objecto criado com a nossa solução e respectiva réplica física, obtida através de impressão 3D.

2. TRABALHO RELACIONADO

O uso de modelos 3D em aplicações de computação gráfica tem crescido ao longo dos anos, assim como a necessidade de desenvolver ferramentas para os criar e modificar. Uma ferramenta frequentemente utilizada para a modelação 3D é o Maya da Autodesk¹. Apresenta uma interface utilizador complexa, mas consistente, com atalhos fáceis de compreender. Uma característica importante da sua interface é o *Hotbox*, um menu personalizável e flutuante, cuja visibilidade é rapidamente alternada, e contém todas as ações da aplicação, divididas em subsecções. Outro software de modelação 3D é o SketchUp², com foco em design e modelação para arquitetura, engenharia civil e mecânica. O SketchUp está diretamente ligado com 3D Warehouse, um repositório on-line que permite aos utilizadores pesquisar, armazenar e partilhar modelos 3D gratuitos.

Um exemplo mais clássico de modelação é o SKETCH [Zelevnik 96], no qual o utilizador cria geometrias através de sequências de traços que definem a forma e os seus detalhes. Com três linhas perpendiculares, a entrada é reconhecida como uma caixa com lados definidos pelos tamanhos das linhas. Este método permite ao utilizador criar formas básicas rapidamente, sendo capaz de as alterar mais tarde. O sistema prevê apenas um pequeno conjunto de formas básicas, permitindo que o conjunto de gestos disponíveis possa permanecer relativamente pequeno e icónico das formas que representam. Com uma solução diferente, Teddy [Igarashi 99] baseia-se exclusivamente no traço do utilizador, ao invés de tentar entender a que forma previamente armazenada é que o utilizador se refere. Esta abordagem usa a silhueta 2D desenhada para criar uma versão 3D insuflada dela (uma *blob*). Comparado com o método anterior, a insuflação de *blobs* não é limitada pelo tamanho de conjunto de dados armazenado. Por outro lado, proporciona um menor grau de controlo, sendo mais adequado para formas com faces arredondadas e sem arestas afiadas.

O sistema GIDeS [Fonseca 04] procura melhorar a usabilidade de um sistema de CAD nos estágios iniciais do design de um produto. Para isso, proporciona uma interacção semelhante ao papel e caneta através de uma interface caligráfica, menus dinâmicos e a capacidade de construir desenhos precisos a partir de rascunhos recorrendo restrições. Para manter o utilizador envolvido na modelação, o sistema sugere formas de acordo com o progresso actual, ajudando e fornecendo uma experiência responsiva.

Usando os modelos hierárquicos de volumes implícitos (*BlobTrees*) como uma representação da forma, Schmidt et al. [Schmidt 05] propuseram o ShapeShop, uma ferramenta de modelação de sólidos baseada em esboços e *blobs*, capaz de reproduzir topologias arbitrárias com diversas complexidades. A hierarquia de volume subjacente é usada como uma história da construção, permitindo que os componentes individuais do esboço possam ser não-linearmente editados e removidos. ShapeShop inclui ainda

outras operações baseadas em esboço, como cortar buracos ou adicionar detalhes na superfície. Além da abordagem da insuflação, superfícies podem ser criadas utilizando varrimentos lineares, através de extrusão controlada ou usando superfícies de revolução.

Mais tarde, Lopes et al. [Lopes 11] acrescentaram a possibilidade de utilização de uma caneta digital ao ShapeShop, com o objetivo de alcançar uma experiência mais natural e fluída. Nesta abordagem, os autores delegam as operações de esboço para a mão dominante, usando a caneta, enquanto as operações de manipulação são realizadas com a mão não-dominante através de multi-toque. Esta divisão do trabalho na interacção bimanual foi estudada por Guiard [Guiard 87], o qual descreve a mão dominante como mais apropriada para movimentos finos e manipulação de ferramentas, enquanto que a não dominante pode ser usada para movimentos pouco precisos e definir eixos de referência. Trabalhos mais recentes, como o Mockup Builder [DA12] aproveitam a distinção entre as operações das mãos dominante e não dominante para criar e manipular formas 3D com gestos familiares. Este usa técnicas de interacção em e acima da superfície, permitindo a criação de formas 3D usando gestos de empurrar e puxar, efectuando extrusões de perfis previamente desenhados. Embora tenha alcançado resultados satisfatórios, a aplicação requer uma complexa configuração de hardware.

Com o foco num sistema de criação de modelos LEGO, o LSketchIt [Santos 08] combina interacção caligráfica, um solucionador de restrições e um mecanismo de recuperação 3D. O utilizador é capaz de adicionar um objeto à cena desenhando-o. Este esboço é então usado numa interrogação, permitindo que o sistema ofereça uma lista de sugestões em que o utilizador pode seleccionar o objecto a ser usado. A interface caligráfica permite ainda que o utilizador desenhe detalhes nas peças recuperadas, possibilitando o refinamento da pesquisa.

Visto ser difícil modelar precisamente objetos 3D quando estes têm características exactas como simetrias ou subestruturas repetidas usando apenas interfaces gestuais, Igarashi e Hughes [Igarashi 01] desenvolveram uma interface sugestiva para desenho 3D, como parte integrante do Chateau, um sistema de modelação 3D. Quando uma forma é desenhada, o sistema procura inferir possíveis operações seguintes e apresenta-as numa lista de miniaturas. Este método de previsão e apresentação de diferentes resultados possíveis é muitas vezes referida como lista de sugestões ou de expectativas. Quando se lida com um conjunto de formas complexas num cenário de recuperação de objectos 3D, é provável que estas sejam pouco diferentes umas das outras, levando a um maior número de ambiguidades ao analisar a entrada do utilizador. Para resolver este problema, Fonseca et al. [Fonseca 04] e Santos et al. [Santos 08] exploraram o conceito de listas de sugestões como um menu dinâmico não-intrusivo, que é acionado quando os traços do utilizador são reconhecidos ambigualmente. Schmidt et al. [Schmidt 05] também usaram listas de sugestões no ShapeShop para resolver ambiguidades.

¹Maya, www.autodesk.com/products/maya/

²SketchUp, www.sketchup.com/

3 APLICAÇÃO PARA MODELAÇÃO 3D

Com o objectivo de criar uma solução para modelação 3D de fácil utilização e aprendizagem, desenvolvemos uma aplicação web que corre num *browser*, apresentada na Figura 2, tornado-a acessível a computadores com e sem ecrã táctil e tablets. Com foco numa interação mais natural, implementámos ainda três técnicas diferentes de interacção que tiram proveito de tecnologias multi-toque, caneta digital e uma combinação de ambas. Nas próximas subsecções descreveremos as funcionalidades de modelação e manipulação de câmara e objetos. Detalhes específicos da interacção serão descritos mais à frente.

3.1 Modelação

Dado que o nosso objetivo era uma solução que possa tirar proveito tanto de multi-toque como de interacção com caneta, foi considerada uma abordagem baseada em esboços para a modelação 3D. À semelhança de outros trabalhos [Igarashi 99, Igarashi 01, Fonseca 04, Santos 08], o processo de modelação baseia-se na captura do desenho livre do utilizador, que é analisado para determinar o resultado desejado. Na nossa abordagem, sempre que um utilizador termina um traço, os possíveis resultados são calculados, podendo o utilizador escolher uma possível transformação através de uma lista de sugestões.

Formas Primitivas: Ao desenhar um ou mais traços, o utilizador é capaz de compor uma forma 2D que pode mais tarde ser utilizada para a criação de um modelo 3D com características semelhantes. Para o fazer, o traço 2D desenhado é projetado sobre o plano base da cena, posicionando o traço no espaço 3D, para criar o novo modelo. Assim, o propósito principal do traço 2D é definir os limites geométricos dos objetos 3D.

Extrusão Giratória: Usando este método, o utilizador é capaz de criar formas extrudidas complexas com elevado nível de controle (Figura 3) com base num perfil 2D fechado. O utilizador pode controlar a extrusão usando duas áreas distintas do ecrã. Tanto na zona lateral esquerda como direita do ecrã (cerca de 20% do tamanho do ecrã

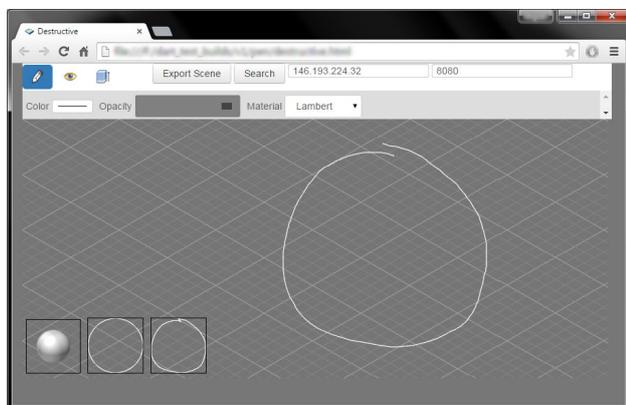


Figura 2. Ecrã da aplicação desenvolvida, com lista de sugestões para um esboço.

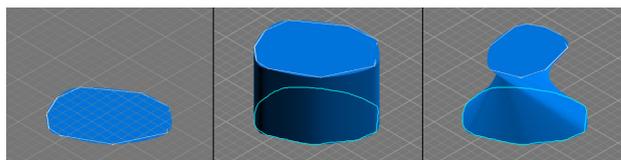


Figura 3. Um modelo criado através de extrusão giratória. A partir de um perfil fechado é possível extrudi-lo e manipulá-lo.

de cada lado) a entrada do utilizar serve para definir a altura (eixo Z) da extrusão. No centro do ecrã, o utilizador pode realizar transformações no perfil 2D fechado. Essas transformações incluem rotação, escala e translação (no plano XY), e permitem a criação de modelos complexos. As três transformações do perfil podem ser individualmente ligadas ou desligadas, uma vez que controlar todas em simultâneo pode ser complexo.

3.2 Manipulação da Câmara

A fim de permitir uma experiência satisfatória e fluida num contexto de modelação 3D, é crucial fornecer controlos naturais da câmara. Para permitir ao utilizador navegar livremente, oferecemos três manipulações diferentes: translação, *zoom* e órbita. A translação (*pan*) consiste em mover a câmara de forma paralela ao plano de visualização. O *zoom* é alcançado pela diminuição ou aumento do campo de visão (*FOV*) da câmara, exibindo uma parte maior ou menor da cena e, portanto, criando a ilusão de que a câmara se aproxima ou afasta. Finalmente, a órbita da câmara permite rodar em torno de um determinado ponto no espaço 3D. Esta manipulação só permite que a direcção da câmara esteja num ângulo entre 15 e 90 graus em relação ao chão da cena. Ângulos abaixo de 15 graus podem causar problemas visto que a direcção da câmara se torna quase paralela ao chão da cena, enquanto ângulos verticais acima de 90 graus irão virar desnecessariamente a câmara. A manipulação da câmara ocorre quando um gesto específico é aplicado directamente na cena, e não num qualquer objeto existente seleccionado.

3.3 Manipulação de Objectos

Para compor modelos complexos, a nossa solução permite a manipulação básica de modelos individuais. Nesta secção iremos descrever informações genéricas para manipulação de objetos, enquanto nas secções seguintes descreveremos interacções específicas em relação a cada tipo de entrada. Um objeto pode ser transformado através de translação no espaço 3D, escala simples (proporcional nos três eixos) e rotação em cada eixo individualmente. Para efeitos de coerência, estas transformações são análogas às transformações da câmara: *pan*, *zoom* e órbita, respectivamente. Para remover ambiguidades, a manipulação de um objecto só pode ser feita se o gesto lhe for aplicado e o próprio objecto esteja seleccionado.

Como informação de entrada do utilizador é baseada em posições 2D do ecrã, esta não contém informação sufi-

ciente para descrever transformações nos três eixos do espaço 3D. Assim, para a nossa solução escolhemos definir transformações de translação e rotação dependentes da orientação da câmara (similar a [Mendes 11]). Ao fazê-lo, em qualquer altura, um objeto só pode ser movido ou rodado num máximo de dois eixos, decididos tendo em conta a orientação actual da câmara. Por exemplo, se o sentido de câmara está mais alinhado com o eixo X, então os eixos a ser considerados para translações serão Y e Z e para rotações só X. Esta abordagem não só evita sobrecarregar o utilizador com menus extras ou combinações complicadas de entrada, como também permite posicionamento e rotação precisos de uma forma familiar.

3.4 Listas de Sugestões

Como comprovado noutros trabalhos [Igarashi 01, Schmidt 05, Fonseca 04], durante o processo de modelação é importante manter o utilizador sempre produtivamente empenhado, fornecendo as ferramentas adequadas conforme são necessárias. Para isso, listas de sugestões desempenham um papel fulcral na nossa solução. Recorremos a reconhecedores que procuram activamente transformações possíveis dado o estado actual da aplicação. Estas possíveis modificações são usadas para preencher uma lista de sugestões, através de miniaturas localizadas no canto inferior esquerdo da janela. Cada elemento da lista contém uma visualização tridimensional animada do respectivo resultado. A lista é não-invasiva, sendo que o utilizador pode simplesmente ignorar as sugestões sem qualquer repercussão. É importante notar que as listas de sugestões não devem intimidar o utilizador com muitos resultados, ainda mais visto que a sua maioria provavelmente não será útil no contexto actual.

Os resultados de interrogações para recuperação também podem resultar em resultados para a lista de sugestões. À medida que o utilizador compõe um modelo mais complexo utilizando outras formas mais simples, estas podem ser agrupados como um único objeto a ser utilizado para recuperação. Assim que o servidor de recuperação responde com resultados, as sugestões são preenchidas com os objectos que mais se assemelham.

3.5 Recuperação de Objectos 3D

Como é difícil produzir formas exactas utilizando esboços, a integração com um sistema de recuperação do objecto 3D permite ao utilizador especificar um objecto relacionado com uma forma desejada e consultar uma base de dados usando o esboço como exemplo. Assim é possível adquirir formas complexas modelando apenas uma aproximação. Este processo não só é mais rápido, como mais simples. Por outro lado, é limitado pelo conjunto de formas armazenadas no sistema, bem como pelos descritores usados. Embora o componente de recuperação seja considerado uma característica importante da solução, dado ainda não ser altamente preciso na medição de valores de similaridade entre os modelos, decidimos remover a submissão automática para recuperação. Em vez disso, o utilizador pode seleccionar objectos para executar uma consulta sempre que considerar apropriado.

4 ESTILOS DE INTERACÇÃO

Embora o foco deste trabalho seja o desenvolvimento de uma nova técnica para modelação 3D baseada em esboços e combinando caneta digital e multi-toque, é importante conceber e avaliar outras técnicas para medir e comparar as suas vantagens e desvantagens. Para isso, desenvolvemos inicialmente uma interacção simples baseada em caneta, usando um único ponto de contacto, de forma semelhante a outros trabalhos [Igarashi 01, Schmidt 05].

4.1 Interacção Baseada em Caneta

Nesta abordagem concentrámo-nos em definir claramente as diferentes etapas do processo de modelação, apresentando ao utilizador os menus principais de desenho, extrusão giratória e manipulação. Na secção de desenho o utilizador é capaz de desenhar traços livremente, podendo recorrer à lista de sugestões para transformar as formas desenhadas em objectos 3D.

No menu principal de manipulação (apresentado na Figura 4), o utilizador é capaz de seleccionar e manipular objectos e câmara. Por defeito, as definições da câmara e dos objetos estão definidas para *pan* e translação. Isto permite ao utilizador mover livremente um objeto seleccionado no plano designado (definido pela orientação da câmara) fazendo um traço a partir da localização do objecto para a posição final desejada. A execução do mesmo gesto partindo da cena ou de um objecto não seleccionado executa a operação de *pan*. A (des)selecção de um objecto é executada ao tocar num objeto. Ainda no menu principal de manipulação, é possível alterar as operações para manipulação da câmara e dos objectos.

Para extrusão giratória, nas secções laterais do ecrã é possível realizar traços verticais para aumentar ou diminuir o nível da extrusão. Na restante área central do ecrã, o movimento vertical da caneta afecta a dimensão do perfil de extrusão, enquanto que o movimento horizontal gira o perfil no eixo Z (paralelo com o plano base da cena). Um *tap* com a caneta termina a extrusão e finaliza a forma.

4.2 Interacção Baseada em Toque

Também é importante medir o desempenho de uma abordagem baseada exclusivamente em multi-toque. Portanto, uma abordagem assim foi desenvolvida com os mesmos menus principais utilizados na abordagem baseada em caneta, por forma a isolar as funcionalidades de desenho, manipulação de câmara e objectos, e extrusão giratória.



Figura 4. Menus da abordagem com caneta: modos de desenho, manipulação e extrusão (vermelho), manipulações da câmara (verde) e de objectos (amarelo).

Enquanto no modo de desenho, a modelação com esboços pode ser feita por meio de um ou mais dedos simultaneamente. No modo de manipulação, é possível navegar na cena e manipular objectos usando gestos específicos.

Os gestos de manipulação de objetos estão exemplificados na Figura 5. Antes de um objecto poder ser manipulado, tem de ser seleccionado através da realização de um toque simples (*tap*) sobre ele. Para aplicar uma transformação de translação a um objeto, um gesto de arrastar com um único toque (a partir do objeto) permite que ele seja movido no espaço. A rotação e escala têm lugar assim que são feitos dois toques no objecto. Alterações tanto na distância como no ângulo entre estes determinam a escala e a rotação, respectivamente. O tamanho do objecto varia directamente com esta distância. O objecto é rodado no eixo mais próximo da orientação da câmara.

Quando os gestos de manipulação são aplicados na cena ou a objectos não seleccionados, a câmara é manipulada. Os gestos de manipulação da câmara estão na Figura 6. Para fazer *pan* da câmara, o arrastar de um toque moverá a câmara paralelamente à base da cena, mantendo a sua altura. Com dois toques simultâneos é possível mudar o *zoom* e órbita da câmara. Um ponto médio é calculado e utilizado para projectar um ponto no plano XY, fornecendo um ponto central para as transformações. Para o *zoom*, as variações na distância dos toques determinam se a câmara se aproxima ou afasta do ponto projectado. Para a órbita, o ponto projectado actua como centro, o que significa que no final a câmara continuará para aí virada.

Tendo um esboço fechado e mudando para o menu principal de extrusão giratória inicializa o método. Neste estado, o controlo do nível de extrusão pode ser feito com um único toque ao longo do lado esquerdo ou direito da janela da aplicação. Na área central restante, o utilizador é capaz de, com dois toques, modificar o perfil da extrusão. As alterações no ângulo e na distância dos toques alteram a escala e a rotação (no eixo Z) do perfil. Um *tap* na extrusão em curso finaliza o seu progresso.

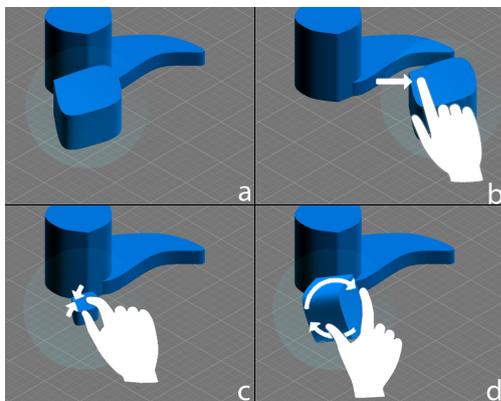


Figura 5. Manipulação de objectos com gestos multi-toque: a) objecto seleccionado; b) translação; c) escala; d) rotação.

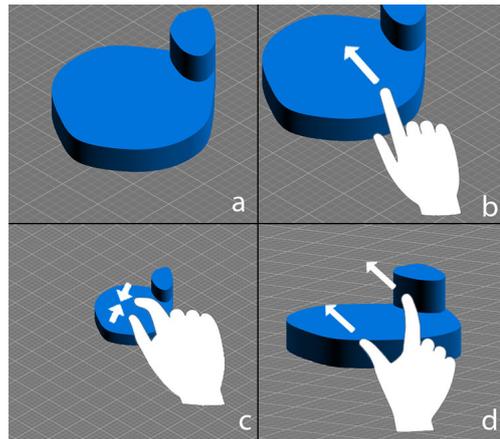


Figura 6. Manipulação da câmara com gestos multi-toque: a) estado inicial; b) pan; c) zoom; d) órbita.

4.3 Interação com Caneta e Toque Combinados

A abordagem que combina caneta e toque visa tirar partido das características mais positivas de ambos os métodos. Nesta abordagem, a interação com caneta é orientada à criação de esboços e ao controlo do nível da extrusão, enquanto transformações e interação com a lista de sugestões são feitas através de toque. Uma vez que as interações com caneta e toque podem ser claramente distinguíveis, nesta abordagem não há menus e todas as acções são realizadas directamente na vista da cena.

O desenho de esboços na abordagem combinada é feito usando a caneta. Como anteriormente, os resultados são mostrados na lista de sugestões e permitem que o utilizador escolha um destes tocando nele. Quanto às manipulações da câmara e de objectos, estas são directamente herdadas da abordagem baseada em toque, recorrendo a um *tap* para (des)seleccionar formas, ao arrastar de um toque para *pan* da câmara e translação de objectos, e a dois toques para *zoom* e órbita da câmara, bem como para escala e rotação de objectos. Sem uma opção de menu para a extrusão giratória, o método é iniciado quando existe um esboço fechado e é realizado um traço para cima com a caneta num dos lados da janela. Ao entrar no modo de extrusão giratória, traços verticais com a caneta ajustam o nível de extrusão, enquanto a interação multi-toque controla as transformações do perfil da extrusão (Figura 7).

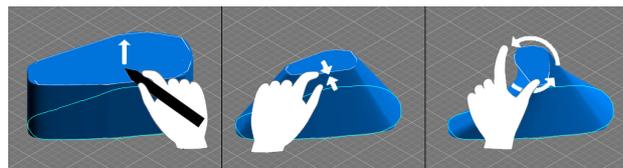


Figura 7. Extrusão giratória na abordagem de caneta e toque. A altura é definida com a caneta e a manipulação do perfil com toque.

5 AVALIAÇÃO COM UTILIZADORES

Tendo sido implementados os três diferentes tipos de interação já descritos, conduzimos uma avaliação com utilizadores para os compararmos. Cada participante executou a mesma tarefa usando as três abordagens, com uma ordem alternada. Para cada sessão, a duração prevista foi de 50 minutos. Antes de qualquer interação com a aplicação, um pequeno vídeo com as funcionalidades básicas do tipo de interação actual foi mostrado aos participantes. Em seguida, estes foram convidados a usar a aplicação numa sessão de habituação com um máximo de 2 minutos. Após executarem a tarefa que lhes era então fornecida, os participantes eram solicitados a preencher um questionário sobre a sua experiência.

A tarefa proposta consistiu na modelação de um candeeiro de mesa (Figura 8.a), composto por cinco sólidos separados: base, coluna, braço, abajur e lâmpada. Cada participante recebeu uma folha de instruções com cinco passos aconselhados, embora não fossem obrigados a segui-los. Da mesma forma, os participantes também não eram obrigados a realizar pesquisas de recuperação, embora tenham sido encorajados a tentar. Para modelar correctamente o candeeiro, os participantes tinham que modelar cilindros simples para a maioria dos componentes (coluna, base e braço), o que poderia ser feito com extrusões. Para a lâmpada, pode-se utilizar uma esfera primitiva. O elemento mais complexo era o abajur, que requer uma extrusão com pelo menos três passos, tendo-se de manipular o perfil da extrusão.

A avaliação com utilizadores contou com a participação de 20 pessoas, a maioria do sexo masculino, com idades entre os 21 e os 57 anos, embora a grande maioria estivesse abaixo dos 30 anos. Em relação ao nível de escolaridade, a maioria dos participantes tinham pelo menos o grau de licenciado. A grande maioria dos participantes tinha vasta experiência com dispositivos multi-toque. Foi utilizado uma mesa interactiva de grandes dimensões, como é visível na Figura 8.b, que conta com uma televisão de 55", tem capacidade de seguir 10 toques e suporta a utilização de uma caneta digital.

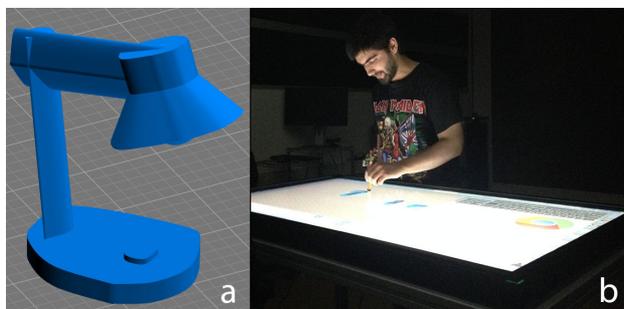


Figura 8. Avaliação com utilizadores: a) candeeiro de mesa a ser modelado; b) participante desenha com caneta na abordagem combinada.

5.1 Dados Objectivos

Tendo sido medido o tempo necessário para cada participante completar a tarefa usando cada tipo de interação, aplicámos o teste de Shapiro-Wilk às amostras, sendo que nem todas seguem uma distribuição normal, pelo que seguimos a análise com testes não paramétricos. O teste de Friedman sugere que houve uma diferença estatisticamente significativa no tempo gasto pelos utilizadores para completar a tarefa usando diferentes abordagens de interação ($\chi^2=17.360$, $p=.001$). Uma análise post hoc com o teste de Wilcoxon Signed-Ranks foi realizada, aplicando a correção de Bonferroni, resultando num nível de significância de $p<.017$. Este teste sugere que existem diferenças estatisticamente significativas ao comparar a abordagem baseada em caneta com a de toque ($Z=-2.495$, $p=.013$) e com a combinada ($Z=-3.461$, $p=.001$), mas tal não se verifica ao comparar as abordagens de toque e combinada. A Figura 9 mostra a visão global dos resultados com os três tipos de interação, notando-se a abordagem baseada exclusivamente em caneta como a mais lenta.

5.2 Dados Subjectivos

Nos questionário, pedimos aos participantes que classificassem a facilidade sentida para realizar a tarefa, bem como para as várias operações de câmara e de objetos, usando uma escala de Likert com 4 pontos (1 - Muito Difícil, 4 - Muito Fácil). Também foi medido o factor diversão (1 - Nada Divertido, 4 - Muito Divertido) e fluidez da abordagem (1 - Nada Fluída, 4 - Muito Fluída). O teste de Wilcoxon Signed-Ranks foi utilizado mais uma vez para determinar se as diferenças encontradas eram estatisticamente significativas. Considerando-se a Tabela 1 podemos assumir que, em geral, os utilizadores consideraram a solução (incluindo todos os três tipos de interações), pelo menos, divertida e fluída. Quanto à facilidade global, os participantes consideraram os três tipos de interação relativamente semelhantes.

Em relação à facilidade de desenhar formas 2D, seleccionar objetos e translação, as diferenças não são signifi-

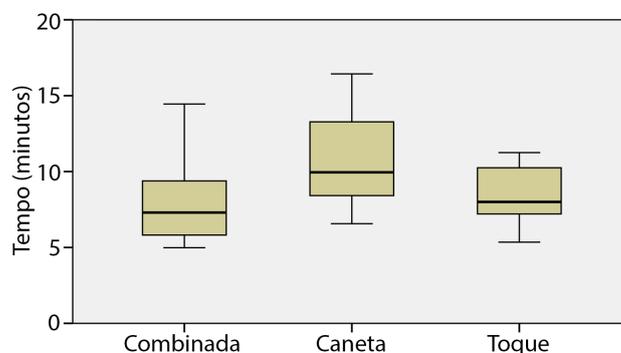


Figura 9. Tempo para completar a tarefa com as três abordagens: mediana, intervalo interquartil (caixas) e intervalo de confiança a 95% (bigodes).

	Caneta	Toque	Combinada
Geral			
Diversão	2.90 (1)	3.25 (1)	3.45 (1)
Facilidade	2.65 (1)	2.65 (1)	2.75 (1)
Fluidez	2.65 (1)	3.00 (0)	3.50 (1)
Desenho	3.75 (1)	3.70 (1)	3.65 (1)
Câmara			
<i>Pan</i>	3.45 (1)	3.15 (1)	3.15 (1)
Órbita	3.10 (1)	2.60 (1)	2.45 (1)
<i>Zoom</i>	3.40 (1)	2.80 (1)	2.65 (1)
Objectos			
Seleção	3.60 (1)	3.50 (1)	3.60 (1)
Translação	3.10 (1)	3.00 (2)	3.10 (1)
Rotação	2.70 (1)	2.35 (1)	2.30 (2)
Escala	3.30 (1)	2.80 (1)	2.55 (1)

Tabela 1. Preferências dos participantes para os diferentes aspectos de cada tipo de interação (média e intervalo inter-quartil). Valores entre 1 (pior) e 4 (melhor).

cativas entre os tipos de interação. No geral, os participantes consideraram tanto o desenho de formas em 2D como a seleção dos objetos operações triviais, enquanto a translação de objectos foi considerada uma tarefa fácil. Para orbitar e fazer *zoom* da câmara, os participantes concordaram fortemente que na interação com caneta são mais fáceis. Orbitar usando a caneta é consideravelmente mais simples do que com apenas toque ($Z=-2.887$, $p=.004$) ou que a abordagem combinada ($Z=-2.982$, $p=.003$). Da mesma forma, o *zoom* também é considerado mais simples com a caneta, do que com toque ($Z=-3.207$, $p=.001$) ou com a combinada ($Z=-3.441$, $p=.001$). Quanto a rodar um objeto, os participantes consideraram todas as abordagens algo difíceis. Por outro lado, redimensionar objectos foi considerada uma operação fácil, tendo sido significativamente melhor na abordagem baseada em caneta do que na de toque ($Z=-2.324$, $p=.020$). Uma vez que a interação baseada em caneta divide as operações de manipulação em diferentes sub-menus, apenas uma operação de manipulação (no máximo) está a ocorrer, em qualquer dado momento. Por outro lado, nos outros dois métodos de interação, várias operações de manipulação podem estar activas num determinado momento, permitindo ao utilizador rodar e escalar simultaneamente um objeto, ou fazer *zoom* e orbitar com a câmara. Da mesma forma, os participantes consideraram a manipulação com caneta mais fácil, uma vez que é mais simples e atómica, embora requeira que se mude os modos de manipulação várias vezes para efectuar uma transformação simples.

Quanto às diferentes ferramentas presentes na solução, a sua utilidade foi classificada com uma escala de 4 pontos (1 - Nada útil, 4 - Muito Útil), apresentadas na Tabela 2. Os participantes concordaram que tanto a ferramenta de extrusão giratória e a lista de sugestões são muito úteis num contexto de modelação 3D. Especificamente para a ferra-

Ferramenta	Utilidade
Extrusão giratória	3.80 (0)
Recuperação de objectos	2.10 (1)
Listas de sugestões	3.70 (1)

Tabela 2. Satisfação dos participantes relativamente às diferentes ferramentas (média e intervalo inter-quartil). Valores entre 1 (pior) e 4 (melhor).

menta de extrusão, esta permitiu a modelação da maioria das peças necessárias para a tarefa do candeeiro de mesa com um número mínimo de interações e com boa precisão. Quanto à lista de sugestões, não só foi considerada crucial para transformar esboços 2D em formas 3D, como também forneceu sugestões úteis para o fecho de traços 2D e apresentação dos resultados de recuperação. Em relação à ferramenta de recuperação de objectos, os participantes consideraram-na pouco útil, dado o facto de que os resultados nem sempre serem os esperados. No entanto, a maioria dos participantes concordou que ter um sistema de recuperação funcional como parte de um aplicativo de modelação 3D pode ser substancialmente útil.

5.3 Observações

No que diz respeito à seleção de objetos, alguns participantes revelaram alguma dificuldade com o facto do objeto ter de ocupar uma quantidade considerável de espaço no ecrã para poder ser tocado. Para contrariar esta situação, um participante sugeriu a criação de uma ferramenta de seleção em caixa, capaz de seleccionar com certeza um ou mais objectos dentro da área.

Ao usar a abordagem combinada de caneta e toque, alguns participantes descreveram as manipulações com uma mão uma tarefa difícil. Estes participantes sentiram-se mais confortáveis colocando a caneta de parte ao fazer manipulações complexas e usando as duas mãos. Em relação às interações baseadas exclusivamente em toque e caneta, os menus foram considerados menos naturais, já que muitas vezes exigiam que se aprendesse que botão estava associado a cada transformação. Além disso, para os métodos baseados em gestos, devido à falta de referências visuais, alguns participantes sentiram inicialmente dificuldades em recordar como executar uma certa acção.

Ao usar a ferramenta de extrusão giratória, um *tap* para finalizar o processo pode ser um problema, uma vez *taps* indesejados podem ocorrer. Foi sugerido a introdução de um menu de confirmação para garantir que o utilizador terminou a extrusão, ou o uso de um gesto mais complexo. Os participantes que escolheram desenhar todas as peças primeiro e montado-as mais tarde foram capazes de realizar a tarefa mais rapidamente. Realizar tarefas semelhantes de forma sequencial requer menos mudanças de menus e de interações com uma e duas mãos.

6 CONCLUSÕES E TRABALHO FUTURO

A fabricação 3D tornou-se um tema popular nos últimos anos, impulsionado pela alta disponibilidade de recursos em 3D online. Além disso, superfícies tácteis são cada vez mais comuns, sendo possível explorar técnicas de interacção novas e mais naturais. Ao longo dos últimos anos, vários trabalhos de modelação 3D foram apresentados, mas nem sempre podem ser generalizadas para outros cenários, nomeadamente para fabricação 3D. Neste trabalho, propusémos-nos identificar vantagens e desvantagens de diferentes tipos de interacção num contexto de modelação 3D, que exploram o uso de superfícies interactivas, com multi-toque e/ou interacção caligráfica. Independentemente do tipo de interacção, pretendeu-se desenvolver uma solução que possa ser utilizada por utilizadores experientes e noviços. Foram implementados três abordagens de interacção: uma com caneta digital, outra com multi-toque e uma que combina as duas anteriores. Para cada uma, diferentes acções são usadas para criar e transformar modelos, bem como para os manipular e ao ponto de vista. As diferentes abordagens de interacção foram avaliadas com utilizadores.

Os resultados revelaram que os participantes foram capazes de realizar a tarefa solicitada mais rapidamente usando a abordagem combinada e, em seguida, a baseada exclusivamente em toque. Esta relação também é consistente com o nível de complexidade dos menus nas abordagens: a abordagem com a caneta usa dois níveis de profundidade, enquanto a de toque tem apenas um, e a combinada não usa menus. Os participantes consideraram fácil e útil a utilização das ferramentas desenvolvidas, como as listas de expectativa, a extrusão giratória e a recuperação de objectos 3D. Com a nossa solução, independentemente de terem experiência prévia com outras ferramentas de modelação, os participantes foram capazes de criar modelos 3D de forma autónoma, prontos para poderem ser posteriormente exportados, armazenadas ou impressos.

Como trabalho futuro, existem alguns detalhes da solução que poderão ser melhorados: recuperação de objectos 3D - embora não sendo o foco do nosso trabalho, este componente é bastante relevante, sendo necessário melhorar o cálculo dos descritores de forma e dos valores de semelhança; gestos multi-toque - como alguns utilizadores apontaram, um simples gesto de tocar para terminar uma extrusão pode levar a que esta fosse concluída involuntariamente, pelo que se deverá encontrar uma alternativa para este gesto; novas ferramentas para modelação - ficou claro que os utilizadores se sentem motivados ao usar ferramentas de criação de formas complexas, como a extrusão giratória. Como seguimento, seria interessante o desenvolvimento de uma ferramenta para criar modelos através de revolução, tirando proveito da interacção multi-toque.

AGRADECIMENTOS

O trabalho apresentado neste artigo foi parcialmente financiado pela Fundação para a Ciência e a Tecnologia (FCT) através dos projectos TECTON-3D

(PTDC/EEI-SII/3154/2012), UID/CEC/50021/2013 e da bolsa SFRH/BD/91372/2012.

Referências

- [DA12] Bruno R. De Araújo, Géry Casiez, e Joaquim A. Jorge. Mockup builder: Direct 3d modeling on and above the surface in a continuous interaction space. Em *Proceedings of Graphics Interface 2012*, GI '12, páginas 173–180, Toronto, Ont., Canada, Canada, 2012. Canadian Information Processing Society.
- [Fonseca 04] Manuel J. Fonseca, Alfredo Ferreira, e Joaquim A. Jorge. Towards 3d modeling using sketches and retrieval. Em *Proceedings of the First Eurographics Conference on Sketch-Based Interfaces and Modeling*, SBM'04, páginas 127–136, Aire-la-Ville, Switzerland, Switzerland, 2004. Eurographics Association.
- [Guiard 87] Yves Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. Em *Journal of Motor Behavior* 19, páginas 486–517, 1987.
- [Igarashi 99] Takeo Igarashi, Satoshi Matsuoka, e Hidehiko Tanaka. Teddy: A sketching interface for 3d freeform design. Em *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, páginas 409–416, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [Igarashi 01] Takeo Igarashi e John F. Hughes. A suggestive interface for 3d drawing. Em *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, UIST '01, páginas 173–181, New York, NY, USA, 2001. ACM.
- [Lopes 11] Pedro Lopes, Daniel Mendes, Bruno Araújo, e Joaquim A. Jorge. Combining bimanual manipulation and pen-based input for 3d modelling. Em *Proceedings of the Eighth Eurographics Symposium on Sketch-Based Interfaces and Modeling*, SBIM '11, páginas 15–22, New York, NY, USA, 2011. ACM.
- [Mendes 11] Daniel Mendes, Pedro Lopes, e Alfredo Ferreira. Hands-on interactive tabletop lego application. Em *Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology*, ACE '11, páginas 19:1–19:8, New York, NY, USA, 2011. ACM.
- [Santos 08] Tiago Santos, Alfredo Ferreira, Filipe Dias, e Manuel J. Fonseca. Using sketches and retrieval to create lego models. Em *Proceedings of the Fifth Eurographics Conference on Sketch-Based Interfaces and Modeling*, SBM'08, páginas 89–96, Aire-la-Ville, Switzerland, Switzerland, 2008. Eurographics Association.
- [Schmidt 05] Ryan Schmidt, Brian Wyvill, Mário Sousa, e Joaquim Jorge. ShapeShop: Sketch-Based Solid Modelling with BlobTrees. *EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling*, 2005.
- [Zelevnik 96] Robert C. Zelevnik, Kenneth P. Herndon, e John F. Hughes. Sketch: An interface for sketching 3d scenes. Em *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, páginas 163–170, New York, NY, USA, 1996. ACM.

Patient Specific Implants Design Tool for Surgeons

Gustavo Mações Nuno Sousa Nelson Alves

Centro de Computação Gráfica
Campus Azurém, Guimarães
{gmacaes, nsousa, nalves}@ccg.pt

Nelson Fânzeres Rui Coelho
Tailored Implant
Rua da Torrinhã, Porto
{nelsonfanzeres, ruicoelho}@tailoredimplant.com

Abstract

One of the major issues in designing Patient Specific Implants (PSI) is the distance between the “language” of surgeons and the one of computer graphics technicians. This gap can induce mistakes in the process of a PSI design. The work described in this article removes the need for a computer graphics technician by implementing a PSI Design Tool that highly automates the PSI design process, and provides a user friendly step-by-step approach. As a result, a surgeon can be responsible for the full process achieving the optimal solution.

Keywords

Patient Specific Implants, Tomography, DICOM, Computer Graphics, 3D Modelling

1. INTRODUCTION

Patient Specific Implants (PSI) are described as the optimal solution for bone and joints replacement [1]. This technique creates implants that are tailored for the specificities of each patient, hence achieving better results. There are several studies focused in the creation of PSI, ranging from tooth reconstructions [2] to the fabrication of complex tissue substitutes [3]. The design of a PSI requires a 3D model of the patient’s area to be treated, the modelling of the PSI and finally the manufacturing of the PSI itself, in a process that is iterative. This process requires medical anatomo-physiology knowledge and 3D modelling skills. The latter is usually assigned to a computer graphics technician (designer), inducing a higher complexity in the process and the possibility of achieving sub-optimal solutions due to differences in professional vocabulary and technical knowledge that can affect communication between the surgeon and the designer.

Currently, several biomedical areas such as dental and limb prostheses are already designed with the help of user friendly softwares that guide the user through the 3D modelling process of a “medical device”. In the case of PSI, the existing market software solutions, like Materialise [4], are expensive and mostly directed to 3D technicians and not for surgeons.

The solution presented in this article aims to create a PSI design tool tailored for surgeons, without the need for 3D

modelling expert skills, giving them full control over the PSI design process. It is expected that this work has a big impact in the PSI quality, allowing surgeons to provide better solutions to their patients, with less invasive techniques and faster healing, meaning quick recovery of patients normal life.

2. PSI DESIGN TOOL FOR SURGEONS

Under this chapter the developed tool for the creation of PSI by surgeons will be described. The purpose of this solution is to enable surgeons to have full control during the whole PSI design process. In order to let surgeons control the full process, the designed solution automates it as much as possible, carefully guiding the surgeon in a step-by-step approach. A user friendly interface was created with only the required functionalities available in each step and minimizing as possible the user inputs, thus, avoiding the need to involve third parties specialized in 3D modelling.

The workflow of the developed PSI tool (Figure 1) is composed by four main steps. First, a set of medical images, generated from a tomography are used to generate a 3D model of the patient’s area to be intervened. Secondly this model is prepared to better suit the PSI creation. Thirdly, the PSI design takes place. Finally, the PSI plus auxiliary data needed to the PSI manufacturing, such as the screws used, are encrypted and exported.

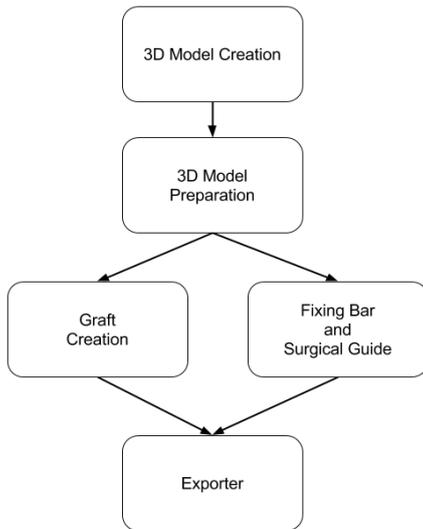


Figure 1 - Workflow of the developed PSI tool

Following sections will define in detail the developed system that, underneath, makes use of these tools: MITK - Medical Image Interaction Toolkit [5], Qt [6] and Blender 3D modelling software [7].

2.1 3D Module Creation

The first stage to design a PSI is the creation of a 3D model of the patient's area to be intervened. This task receives as input a set of medical images in the DICOM format [8] and returns a 3D model.

The user starts by importing the DICOM files to work on, selecting the series to segment. The second step is the segmentation of the regions of interest, in this case bones. It is possible to choose from a list of default types of bones thresholds or to make a custom segmentation (Figure 2) moving a slider that controls the threshold value, while the preview of the expected result is displayed. The 3D model is generated by applying the marching cube algorithm [9] to the segmentation result. In order to achieve a less complex and smoother output a decimation and a median filter are applied to the resulting surface.

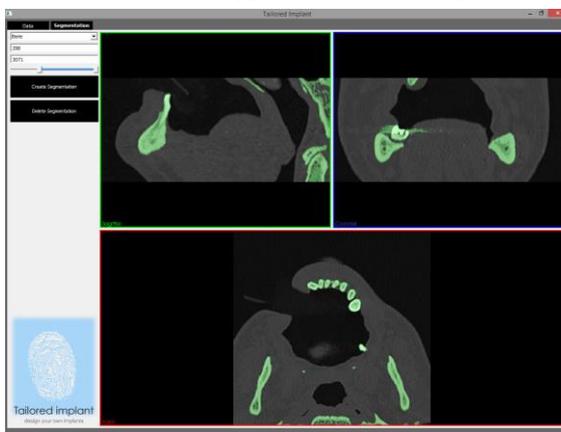


Figure 2 – Dicom Converter

2.2 3D Model Preparation

The generated 3D model from the previous step must be further edited in order to achieve a good working base to the PSI design. The tool allows an adequate visualization of the 3D model, so the user can clearly understand the patient's bone region, thereby avoiding risky areas like the nerves in the jaw. A simplification of the model can also be obtained by reducing it to just the area that needs to be worked. The deletion of parts of the 3D model creates holes in the mesh that need to be closed. An automatic process was implemented to close the mesh. This, involves a pre-preprocessing step, the removal of all the faces that can be built from all the non-manifold edges, which are the faces that cannot be represented in the real 3D world, and may cause conflicts when trying to fill the holes.

With an adequate patient's bone model generated, the user can proceed to the next phase. Two different modules are available at the moment: graft creation or fixing bar and surgical guide creation.

2.3 Design of Grafts

A graft starts with a parallelepiped shape that can be molded to the desired shape. At first, the user adds a new graft and places it in the correct location, orientation and scale against the 3D model of the patient's area to be treated. The graft can then be molded to the desired shape, avoiding hard edges, with a smooth brush tool. Next, automatically, the software removes the graft portion inside the bone, by subtracting them, resulting in a graft that has a perfect cast against the bone surface of the patient's mouth.

The graft implantation in the bone requires the use of screws (see Figure 3). For that purpose, a set of 3D screws models are available. The user adds a new one and position-it correctly against the graft. Once again, the system automatically analyses the intersection of the 3D models and subtracts them, creating a final graft with holes ready for the screws placement.

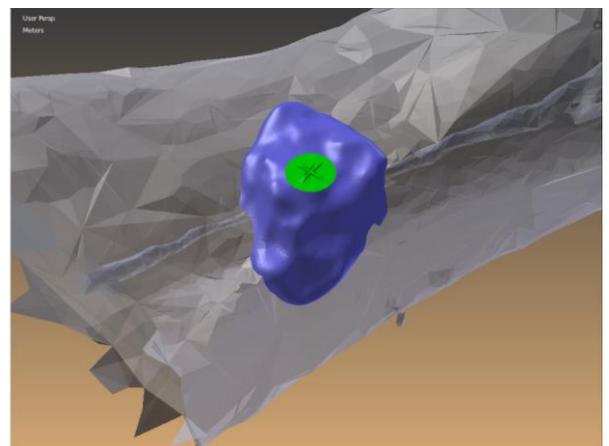


Figure 3 - Graft (blue) and screw (green)

2.4 Fixing Bar and Surgical Guide

If the user decides to produce a fixing bar (presently specific for maxillo-facial treatments) the first step is define the number of implants to import and load them into the scene (see Figure 4). Automatically, the system analyses the jaw of the patient and determines a good initial fit for their position, that can be further adjusted by the user.

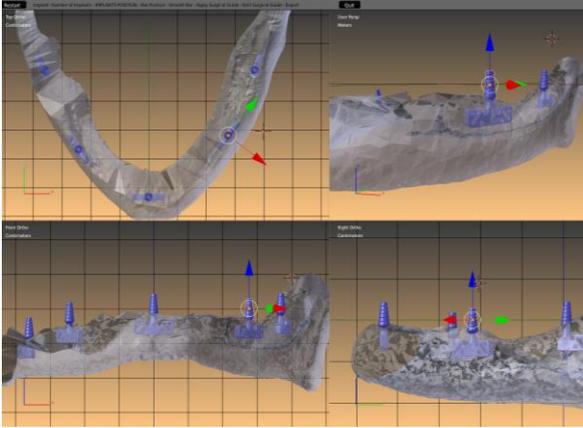


Figure 4 - Positioning Implants

The fixing bar is then automatically created. It will link all the implants that will be placed into the patient mouth. To create the bar, first a Bézier curve with a point in every implant position is generated, then, using a curve modifier tool, a rectangular shape travels along it, resulting in the fixing bar, see Figure 5. Since this is an automatic process, sometimes there is the need to make adjustments to it, so the user can place it correctly, through subdivisions into the main curve that controls the bar.

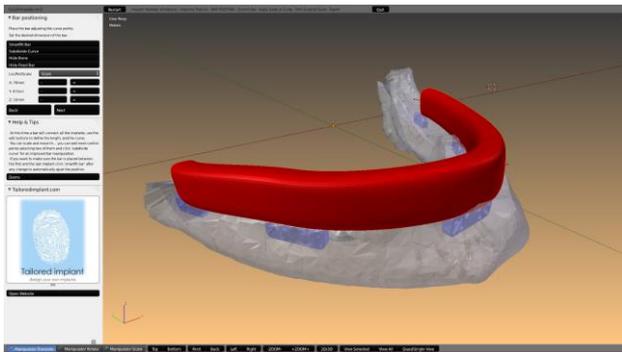


Figure 5 - Fixing bar

Multi-units (structures to fix the fixing bar to the patient's bone) are added to scene and holes in the fixing bar are created to accommodate them.

Finally, a surgical guide is automatically created to assist the surgeon, when making the holes into the bone for pin the fixing bar.

2.5 Exporter

The final stage of the PSI design process is the export of the PSI and auxiliary data necessary for the manufacturing and posterior deploy of it. When exporting, in addi-

tion to the 3D models of the elements that need to be manufactured and a list of used materials, a PDF file (Figure 6) is generated. This stands as an easy and effective way to review the final result, a posteriori, in three dimensions visualization, without the need of using special software. This feature uses both LaTeX [10] scripts and Meshlab [11] engine.

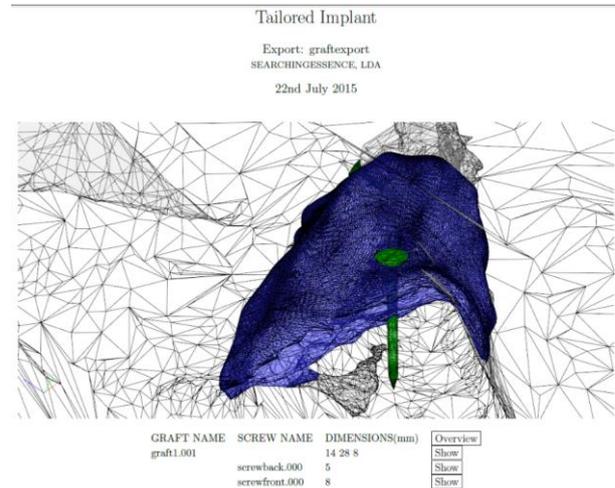


Figure 6 - PDF 3D Guide

The PSI design results are encrypted to guarantee that the outputs can only be used in the original software or accessed by accredited entities. For that purpose, a combination of Advanced Encryption Standard (AES) [12] and Rivest-Shamir-Adleman cryptosystem (RSA) [13] keys is used.

3. CONCLUSIONS

The implementation of a patient specific implants design tool, directed for surgeons, was described in this article. The process of PSI design was highly automated and a comprehensive step-by-step approach was implemented. The core technology has been developed based on many years of experience of a surgeon, and is very close to become available on the market. The feedback from the first tests with real users was very good. There is a real need of a comprehensive platform that allows the surgeons to be closer to the production phase, without having to worry in learning how to work with complex modelling softwares. Currently the solution is more focused in the dental surgery field, and future work will be centered in expanding for others fields like orthopedy.

4. ACKNOWLEDGEMENTS

This work was done by Centro de Computação Gráfica and Searching Essence under the scope of the project, "EASYORTHO - Desenvolvimento de Plataforma B2B para apoio ao desenvolvimento de implantes" project that is co-financed by QREN – I&DT, under the number 33958.

5. REFERENCES

- [1] Yongtae, J. and Kuiwoon C. Design of patient-specific hip implants based on the 3D geometry of the human femur. *Advances in Engineering Software* Volume 41, Issue 4, April 2010, Pages 537–547.
- [2] Yau, H. , Yang, T. and Chen Y Tooth model reconstruction based upon data fusion for orthodontic treatment simulation. *Computers in Biology and Medicine* Volume 48, May 2014, Pages 8–16.
- [3] Sun, W., Starly, B., Nam, J. and Darling A. Bio-CAD modeling and its applications in computer-aided tissue engineering. *Computer-Aided Design* Volume 37, Issue 11, September 2005, Pages 1097–1114.
- [4] Materialise - <http://biomedical.materialise.com/>
- [5] MITK - <http://mitk.org/wiki/MITK>
- [6] Qt - <http://www.qt.io/>
- [7] Blender - <https://www.blender.org/>
- [8] DICOM - <http://medical.nema.org/Dicom/about-DICOM.html>
- [9] Lorensen, W. and Cline, A. Marching cubes: A high resolution 3D surface construction algorithm. *SIGGRAPH '87* Volume 21, Issue 4, July 1987 Pages 163-169.
- [10] LaTeX - <http://latex-project.org/intro.html>
- [11] Meshlab - <http://meshlab.sourceforge.net/>
- [12] ADVANCED ENCRYPTION STANDARD (AES)" (PDF). Federal Information Processing Standards Publication 197. United States National Institute of Standards and Technology (NIST). November 26, 2001.
- [13] Rivest, R. , Shamir A. , and Adleman L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *ACM* Volume 21, Issue 2, Feb. 1978 Pages 120-126.

Mid-Air Manipulation of 3D Models in (Semi-)Immersive Virtual Environments

Vasco Rodrigues Daniel Mendes Alfredo Ferreira Joaquim Jorge

INESC-ID / IST / University of Lisbon

{vasco.t.rodrigues, danielmendes, alfredo.ferreira, jorgej}@tecnico.ulisboa.pt

Abstract

If the interactions made on multi-touch surfaces can be moved to the air above them, a user will have access to a powerful tool. The possibility of tracking the hands' movement in mid-air, creates a world full of interaction possibilities. Despite having been created consolidated solutions for multi-touch interactions, these are limited to a bi-dimensional space, preventing the possibility of direct interactions on the visualization space. Some works were created to try and overcome this problem, however most of them use intrusive techniques, don't use stereoscopy or virtual reality or are lacking accuracy. This work had the objective to explore and evaluate which techniques allow to manipulate with precision 3D objects in mid-air, on a non intrusive semi-immersive/immersive environment. With this work it was possible to observe that the mid-air interactions in immersive systems are the most efficient and satisfying method for users, due to the direct manipulation of 6 degrees of freedom, which mimics the way with which we interact with physical objects.

Keywords

Manipulation, 3D Objects, Precision, Mid-Air.

1. INTRODUCTION

In the recent years, the use of the multi-touch devices has grown, due to the various developments in this kind of technology. From smartphones to tablets, as well as interactive tables, the users have ever more solutions available for their needs. Likewise, the use of 3D objects has largely increased. And so, with the surfacing of the new devices appears a sea of possibilities to interact with the objects.

Multi-touch surfaces have been used in many different types of domains. Due to their configuration, they allow to manipulate virtual objects using the whole hand and not just the fingertips. The possibility of bringing the interactions with the 3D world out of these multi-touch surfaces, using stereoscopy and HMDs, allows to give some immersion degrees to the user.

The 3D object manipulation methods in the tridimensional space have been the subject of studies in recent years. Different approaches have been proposed, however the majority does not give freedom to the user. With the visualization on multi-touch surfaces being on a 2D plane, to manipulate the user is limited to the traditional approaches, applied to a plane. A simple task takes longer to be completed. These problems show a need for a natural, simple and efficient method to interact with 3D objects on multi-touch surfaces.

With this work we wanted to study different techniques to manipulate 3D objects in mid-air, verifying if it is possible

to surpass the existent adversities. Ultimately the preferred technique should allow to overcome the need of a more natural and familiar interaction.

Basing our approach on the set of information found and the defined ideas for the study of non intrusive and natural object manipulation, we developed two systems to test two different manipulation techniques. Using a 3D television, a Kinect camera, a pair of 3D glasses, an HMD Oculus Rift and two Wiimote controllers with a Motion Plus adapter, the user can manipulate immersively and semi-immersively objects. The user interacts with the Wiimotes, viewing the scenario either through the TV and the 3D glasses or through the Oculus Rift. In these prototypes the user has two techniques to interact with the objects. The 6DOF Hands technique allows a direct manipulation of the objects with each of the user's hands so he can move and rotate freely an object as in a real world interaction. There is also the possibility to scale objects using the free hand. The other technique is called Handle Bar and tries to simulate a barbecue spit. The user grabs a bar with both of his hands while the object stands between them. Then the user can manipulate the object by moving the bar like it would move on the real world.

To evaluate the implemented techniques we carried out user test sessions, where we compared both implemented techniques in two different environments, being one of them an immersive setup, and the other one being semi-immersive. The 6DOF Hands technique proved to provide

better precision on the placement of the objects, as well as taking less time to do so. The immersive setup provided better visualization, making the tasks easier, thus being more pleasant to the users.

2 RELATED WORK

Nowadays, the manipulation of 3D objects is used in several applications, being indispensable in several areas. We can easily find the adoption of 3D manipulations in many fields, such as design and assembly of components in engineering and videogames, among others. With the great usefulness that the manipulation of tridimensional objects presents, it has been the subject of several studies and constant development, aided by the improvement and surfacing of new technologies, which turn the tasks easier to complete. Here we describe some of the works related with 3D object manipulation.

Regarding 3D object manipulation on multi-touch surfaces, Cohé et al. [1] created tBox. It allows to manipulate objects with a 3D interaction, taking advantage of a direct and independent 9DOF control. The system allows very precise translations, using a cube whose edges contain the object to manipulate. The object can be moved by dragging the edges and can be scaled by moving away or closer two parallel edges. To rotate the object, one can drag the cube's faces.

Focusing on gesture recognition, Wang et al. [6] created a system that tracks the hands and can recognize a set of gestures in 6DOF. The gestures are created as real world metaphors. It uses a snap system that allows the users to join objects with more precision. The implemented gestures allow to move or even throw an object, instead of dragging it. The great advantage of this system is that it captures the hand gestures without being intrusive, using two Kinect cameras above the interaction space. The background is removed, leaving just the hands, which are then compared to the estimated poses of the gestures on the database. However, it can only be used on a setting where the background can be removed, and with long sleeves.

Following the same line, Hilliges et al. [3] created Holodesk, which combines a transparent screen and a Kinect camera to create the illusion that the user is interacting directly with the virtual environment. The interaction space is located below the screen, giving the impression that the objects are next to the user's hands. The Kinect camera detects the hands and other physical objects' position, allowing to manipulate the virtual objects using them.

De Araùjo et al. [2] developed Mockup Builder, a semi-immersive environment where it is possible to create and manipulate with gestures on and above a surface. It uses stereoscopy, adapting the user's perspective of the 3D environment according to his head position, captured with a Kinect camera. The hands' positions are detected with two Gametrack devices, allowing the user to manipulate objects with 7DOF. By combining the use of two hands with the flexibility of a continuous space, this system allows to change easily between 2D and 3D gestures. The user can

choose the best type of manipulation according to his tasks.

Song et al. [5] proposed the barbecue spit metaphor to control virtual objects in space. They called it Handle Bar, and it tries to mimic a familiar situation to the users: the use of a barbecue spit, being held with both hands and maintaining an object on its center. The user holds an object putting each hand next to the object, moving his hands to move it and rotating it by rotating his hands by the object's center. To scale the object, the user can move his hands away or closer to each other while holding the object. This solution offers a natural way to manipulate the objects while using a familiar situation to the users and being non intrusive.

Still regarding 3D object manipulation, Mendes et al. [4] made a comparison between five different above and on the surface techniques. Using stereoscopy, four techniques are used above the surface, while the fifth uses touch gestures. Above the surface, the developed techniques were 6DOF Hand, 3DOF Hand, Handle Bar (based on the work of Song et al. [5]) and Air TRS. On the surface, the TRS Touch + Widgets technique relies on touching the surface below the object and using widgets. The study concluded that direct interactions above the surface are efficient and satisfactory, while still having problems caused by the lack of precision and hand occlusion. 6DOF Hand was considered more natural to use for reproducing the direct interactions with physical objects, while Handle Bar was as fast as 6DOF Hand.

3 IMPLEMENTATION

The objective of this work was to study natural interaction techniques above an interactive surface. We start by presenting the implemented system used to test the interaction techniques. We also present the implemented manipulation techniques that allow a simple direct interaction, as well as solutions to control the camera on the 3D environment.

3.1 Hardware Configuration

The developed solution uses a Microsoft Kinect 2 camera to track the movement of the user's hands and head. In order to interact with the 3D objects, the user has two Wiimote controllers with a Motion Plus adapter. This wireless device, usually used to play games, has an adapter that allows to obtain the user's hands' orientation. The buttons of the controller allow the user to use the manipulation techniques to interact with the objects.

Another device used with our solution is the HMD Oculus Rift 2, that allows an immersive visualization of the environment. With the help of the Kinect camera, it's possible to detect if the user has changed his position and/or his orientation, updating his perspective and position in the virtual world.

As for the stereoscopic visualization, a combination of a Samsung 3D television and a pair of Samsung SSG-5100GB 3D glasses is used. This setup allows a semi-immersive visualization above the surface. It's possible to view the used devices in Figure 1.

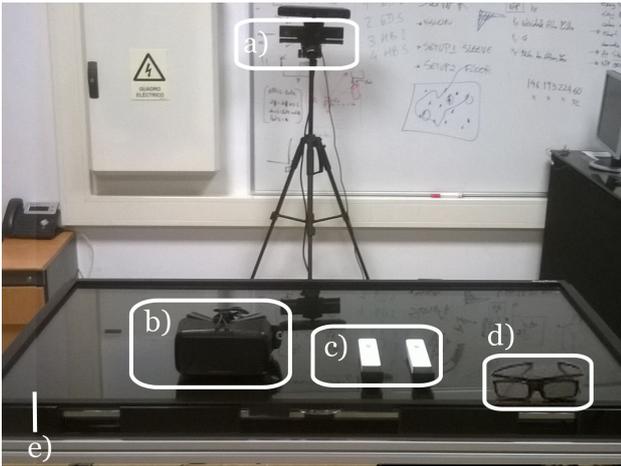


Figure 1. Used devices with the system: a) Kinect 2; b) Oculus Rift 2; c) Wiimote controllers; d) Samsung SSG-5100GB 3D glasses and e) 3D television.

3.2 Architecture

With this work we wanted to create an environment that would allow to find more natural tridimensional manipulations. With this goal in mind, we developed the architecture represented in Figure 2. The system consists in five different modules, with two of them being the core of the architecture.

3.2.1 Gestures Module

The Gestures Module captures the user's movements using a Kinect 2 camera. This data is sent to the module through UDP, which then maps it to the virtual world, recreating the movements virtually in real time. The data contains the user's hands' position.

3.2.2 Interaction Module

The Interaction Module is a pillar of the system. This module allows to use the implemented manipulation techniques. The data received from the Kinect and the Wiimote controllers from the Gestures Module include the buttons' statuses as well as their orientation, given by the

gyroscopes in the Motion Plus adapters, and position. This data allows to recreate the manipulation techniques, grabbing the objects to move, rotate and scale them.

3.2.3 Scene Module

The Scene Module is responsible for dealing with the logic that allows to represent any element in the 3D space, as well as its movements. It allows to show the result of moving an object, rotate it and scale it. It also allows to move through a scene with ease and show different initial states for each. It communicates directly with the Render Module, mapping the visualization in Oculus Rift and the stereoscopic surface.

3.2.4 Viewpoint Module and Render Module

The Viewpoint Module uses the Kinect data about the user's head position to calculate its position on the virtual world. These calculations are sent to the Render Module, which can adapt what is seen by the user on the virtual scene.

In this manner, the Render Module shows the user the virtual environment where he interacts, using Unity 3D to create the virtual scenarios.

3.3 Interaction Techniques

In this work we implemented two interaction techniques which allow the user to manipulate 3D virtual objects, using stereoscopy or virtual reality. The interaction is made in the space above an interactive surface. Based on the works of Wang et al. [6], Song et al. [5] and Mendes et al. [4], the chosen techniques were 6DOF Hands and Handle Bar. These techniques allow a familiar manipulation interaction, having better performance than other studied techniques.

These techniques allow the user to interact in 7 DOF: 3 DOF to translate, 3 DOF to rotate and 1 DOF to scale. In each technique the user can choose which hand to use as dominant to manipulate the object. Since that for each technique the user needs both hands to manipulate in 7 DOF, only one object can be manipulated at a time. We present these techniques with more detail on the next sections.

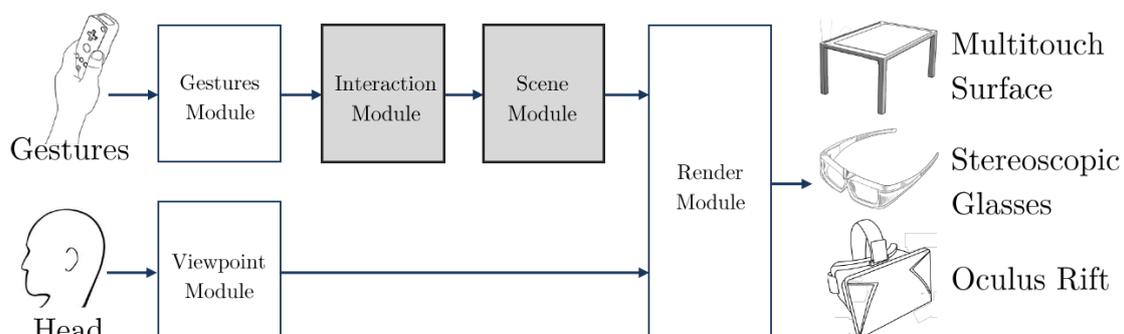


Figure 2. System's architecture.

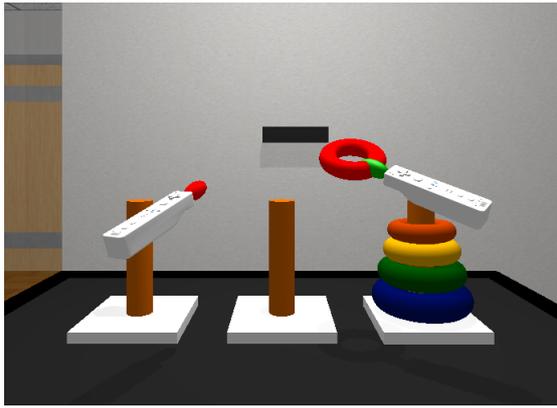


Figure 3. Manipulation using 6DOF Hands.

3.3.1 6DOF Hands

In 6DOF Hands, the user's hands are tracked in real time, allowing to simulate the way one holds, moves and rotates an object with one hand in the physical world. As one can see by the name, one hand allows 6 DOF, three to move and three to rotate. This technique is based mainly on the work of Wang et al. [6].

The user touches with his Wiimote controller the object he wants to interact with, holding it with the B button. Then, the object is moved by following the movements of the hand that holds it. The rotation is detected by the controller, as the user rotates his wrist.

To scale the object, the user uses his other controller. The hand does not need to be touching the object. The scaling is indirect, using the distance between the hands and mapping hands being closer or further away with reducing or enlarging the object's size. We can see an interaction using 6DOF Hands in Figure 3.

3.3.2 Handle Bar

Based on the work of Song et al. [5], we adopted the Handle Bar metaphor. This simulates the use of a barbecue spit to grill chicken. The user holds the spit with both hands, whereas the chicken is placed at its midpoint.

The user can grab and manipulate objects using the midpoint of his hands. This point is virtually represented as a blue dot. To hold the object, the user puts this dot on it, grabbing with both hands. With the object being held like a bar passing through it, the user can manipulate it. He can move and rotate the object, which will mimic the the bar movements. To scale the object, the distance between both hands is used. We can see an interaction using Handle Bar in Figure 4.

4 USER EVALUATION

In order to compare the implemented techniques, it was necessary to conduct a set of user tests. The techniques were evaluated in regards to ease of usage, precision and time. Two systems were developed to allow the comparison immersively and semi-immersively. Each user per-

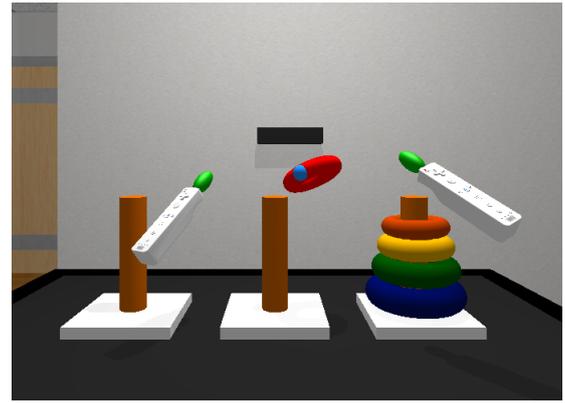


Figure 4. Manipulation using Handle Bar.

formed three placement tasks with each technique in each system.

4.1 Systems

Two different systems were created to test the implemented techniques. One represents an immersive virtual environment (IVE), and the other a semi-immersive virtual environment (SIVE). Each system has a collisions mechanism that detects when a controller is touching a graspable object. The user must touch the objects with an ellipsoid placed in front of the controllers in order to grab them. When the ellipsoid is touching a graspable object, a green wireframe appears surrounding the object.

None of the techniques has physics, to allow a completely free manipulation, which also helps in measuring the precision of the tasks. After each task, the relevant data is automatically saved to a file, allowing its review later.

4.1.1 Immersive Virtual Environment - IVE

The IVE system is a virtual recreation of the room where the tests take place. The goal is to test manipulations with 3D objects, giving the illusion that the user is inside the virtual world.

The user is able to look in each direction, with the help of the orientation given by the Oculus Rift HMD. He can also move on the scene, as the Kinect detects his head's position and applies the needed transformations. This gives freedom to his movements. The user's arms are also recreated using the position data from the Kinect, in an attempt to give a bigger sense of presence in the virtual world.

4.1.2 Semi-Immersive Virtual Environment - SIVE

The environment in the SIVE system consists of a plane that overlaps the 3D television's surface. This allows to use stereoscopy, giving the illusion to the user that the objects are above the surface.

By seeing the objects above the surface, the user can interact with them using the Wiimote controllers. The user's perspective of the scene is given by his head's position with the help of the Kinect camera.

4.2 Methodology

The tests were structured accordingly to the following scheme:

- **Introduction** When starting a new user test, each system was presented, as well as its capacities. It was also explained how to interact with each prototype and the technological limitations present in monitoring the user's hands.
- **Training Session** It was shown to the user how to move, rotate and scale objects with each technique in each system. We allowed the user to understand how to interact and train his interactions.
- **Tasks Execution** We randomly determined which system and technique was the first to be used by the user, to guarantee that the order of usage did not influence the results. It was asked to each user to execute a set of three equal tasks with both techniques in each system. The order of the tasks was predetermined, with an increasing difficulty.
- **Final Questionnaire** After the test sessions, the participants filled in a questionnaire to evaluate different aspects on each technique and system, such as ease of usage, preference and fun factor.
- **Profile Questionnaire** Finally, the users filled in a questionnaire with the objective of getting to know their profile: age and experience towards stereoscopy and other devices.

4.3 Tasks

Knowing that solids are present in one's life early, it was decided that the tasks would follow the idea of fitting objects in holes with its shape. The tasks had incremental difficulty, starting with a simple task and ending with one that needs a better control of the technique to be concluded. As a result, there was no gravity nor physical collision between objects. Also, the user can only manipulate the object that will be fit in the hole. The tasks were the same in each technique-system pair.

The users had a training session for each technique-system. In the scenario the user had a replica of the Tower of Hanoi game, as we can see in Figure 5. With this, the user could manipulate the rings freely and learn how to interact in each technique and system.

During the tests, the tasks consisted in fitting the green plug in the red socket, with the highest possible precision. Only the green plug can be manipulated. In each system the tasks were executed by incremental difficulty. On the first task, the user moved the plug, on the second task the user needed to move and rotate it, and on the third the user needed to move, rotate and scale the plug to achieve the final position. In Figure 6 we can see the initial state of the tasks.

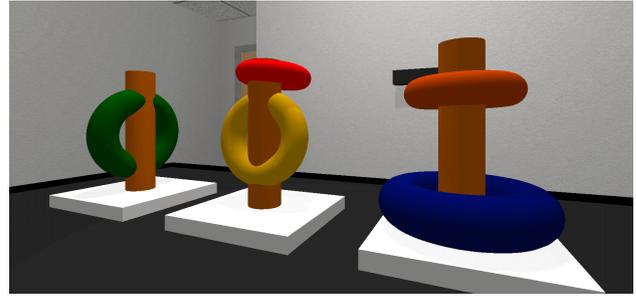


Figure 5. Task performed during the training session.

4.4 Apparatus and Participants

The test sessions were performed on a closed environment, without the possibility of external influences. It was used a Kinect 2 camera, two Wiimote controllers with Motion Plus adapters, 3D Samsung glasses and the HMD Oculus Rift. Both computers used by each system had a Windows 7 operating system, with an Intel Core i7 - 3770K 3.5 GHz CPU and 16 GB RAM.

The evaluation had 20 users, one female and nineteen males, with ages between 19 and 35. Only four users hadn't tried stereoscopy, and eight never had experienced virtual reality. Nineteen of the users had already played with a gaming console, with ten having used the Kinect, fourteen the Wiimote and eight the Playstation Move. On Figure 7 we can see users interacting with each system.

4.5 Results and Discussion

This evaluation allowed us to understand the technique that felt most natural to the users. We analysed our measures using the Wilcoxon Signed-Rank Test, comparing precision and time spent by each user to conclude a task.

4.5.1 Task 1

The difference from the ideal angle of the plug was smaller in 6DOF Hands - IVE than on Handle Bar - IVE ($Z = -2.725, p = 0.006$), as it was easier to rotate an object in 6DOF Hands. Handle Bar has an axis which can't rotate. The time to complete the task was smaller on 6DOF Hands - IVE than on 6DOF Hands - SIVE ($Z = -2.763, p = 0.006$). With the ability to see many different angles using

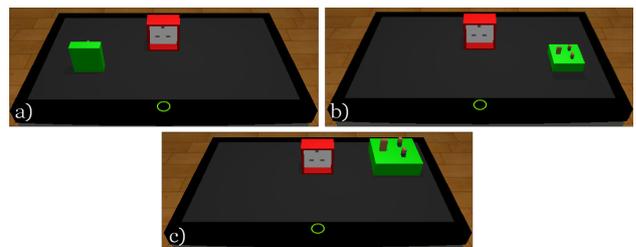


Figure 6. Initial state of the tasks: a) Task 1; b) Task 2; c) Task 3.

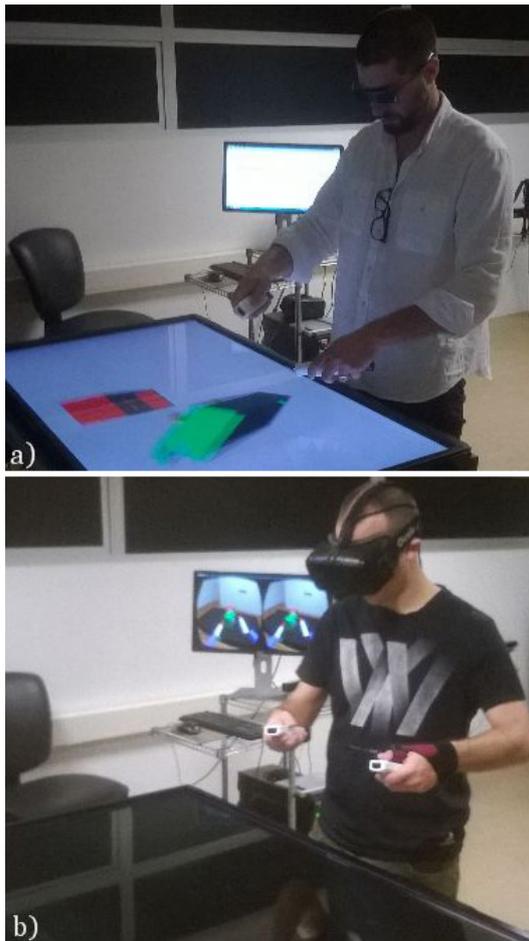


Figure 7. Users testing the SIVE (a) and IVE (b).

the Oculus Rift, it's faster to place an object. In Figure 8 we can see graphs about the measures obtained.

4.5.2 Task 2

The distance from the ideal position of the plug was less with 6DOF Hands - IVE than with Handle Bar - IVE ($Z = -2.576$, $p = 0.01$). That might be caused by the need of rotating the plug, and the lack of a rotation in Handle Bar might tire and bore the users, making them placing the plug with less accuracy. The difference from the ideal angle of the plug was smaller in 6DOF Hands - IVE than in Handle Bar - IVE ($Z = -2.978$, $p = 0.003$), being also smaller in 6DOF Hands - SIVE than in Handle Bar - SIVE ($Z = -2.203$, $p = 0.028$). The task was completed faster with 6DOF Hands - IVE than with Handle Bar - IVE ($Z = -2.800$, $p = 0.005$) and 6DOF Hands - SIVE than with Handle Bar - SIVE ($Z = -3.360$, $p = 0.001$). In Figure 9 we can see graphs about the measures obtained.

4.5.3 Task 3

The distance from the ideal angle of the plug was less with 6DOF Hands - SIVE than with Handle Bar - SIVE ($Z = -2.016$, $p = 0.044$). The rotation difficulty with Handle Bar is even more noticed in the semi-immersive system, as it's more difficult to the user to see the objects

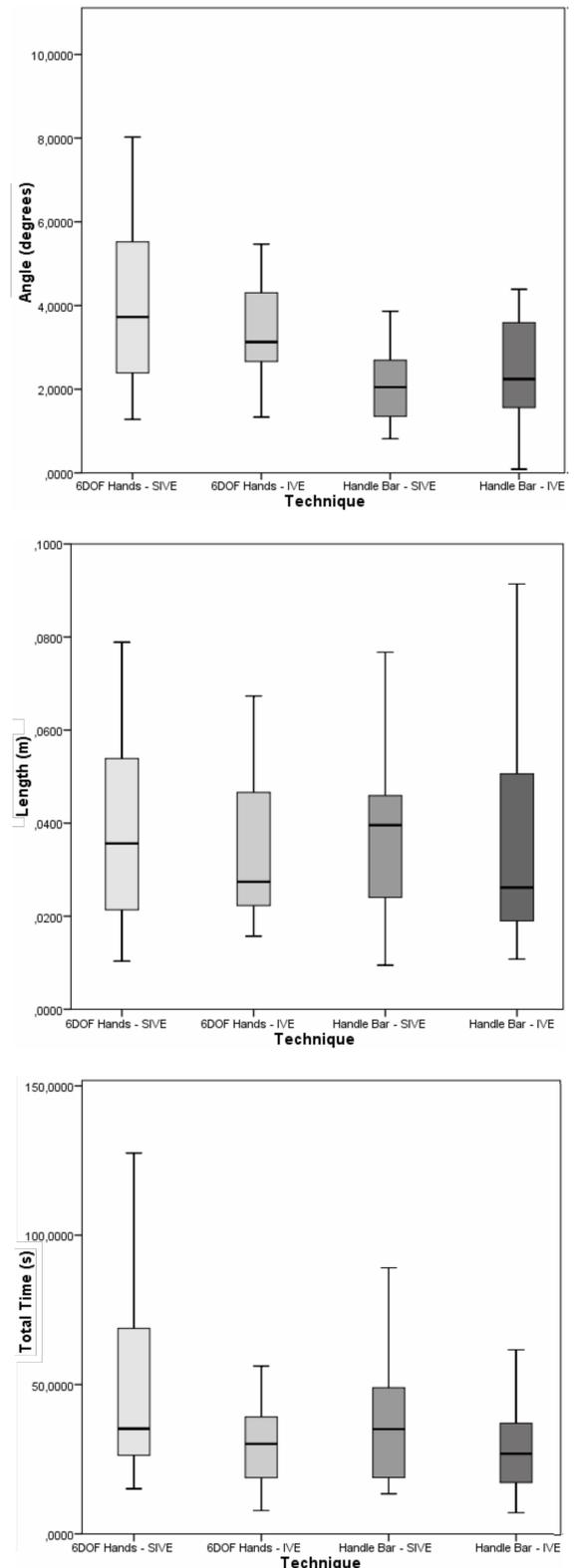


Figure 8. Measures obtained in task 1.

from every angle, as well as understand their true position. Being this the third task of the users with each technique-system pair, the users were able to fit the plug with ease in the same position. In Figure 10 we can see graphs about the measures obtained.

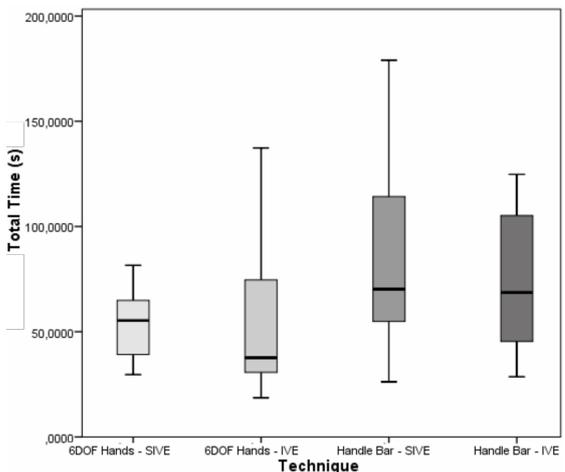
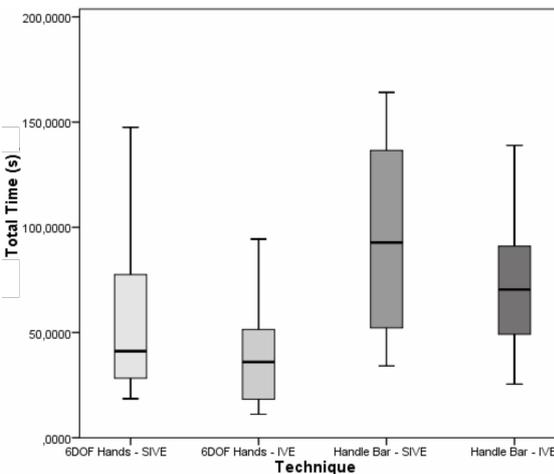
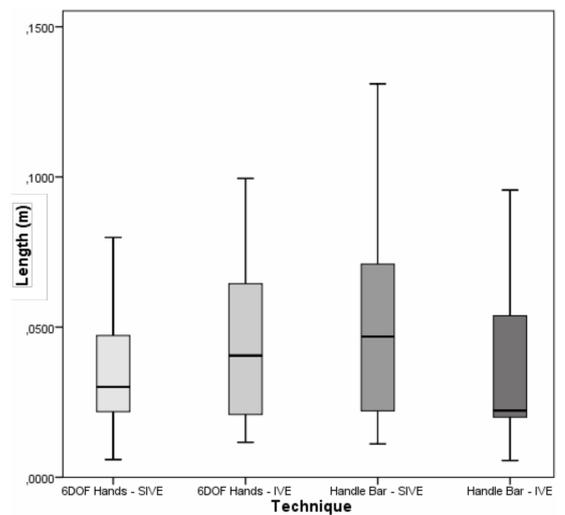
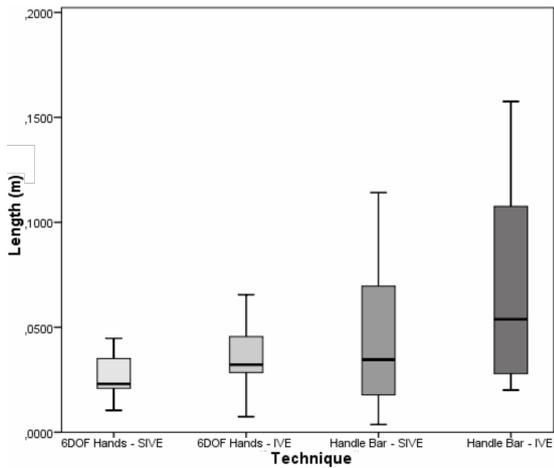
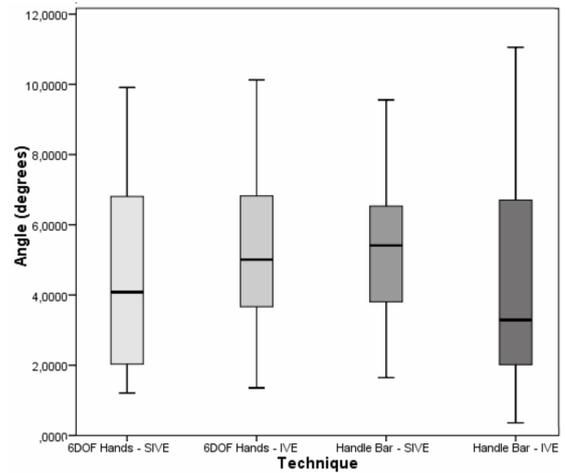
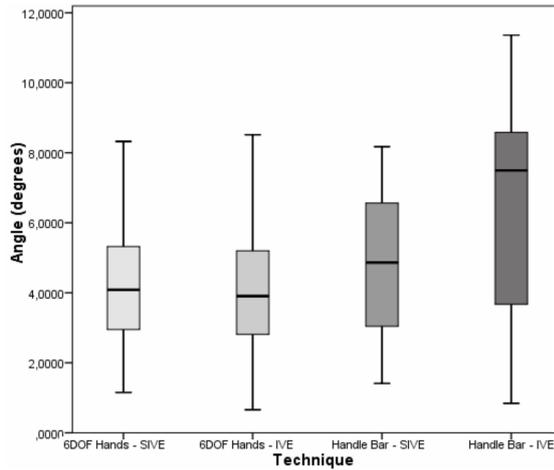


Figure 9. Measures obtained in task 2.

Figure 10. Measures obtained in task 3.

4.5.4 Subjective Data

On the final questionnaire, the users were asked to classify their experience with each technique and system. The users thought that learning how to interact with 6DOF Hands - IVE was easier than with Handle Bar - IVE, 6DOF - SIVE and Handle Bar - SIVE, globally ($Z = -3.220, p = 0.001$ and $Z = -3.066, p = 0.002$ and $Z = -3.440, p = 0.001$) and in the rotation ($Z = -3.779, p < 0.001$

and $Z = -3.542, p < 0.001$ and $Z = -3.495, p < 0.001$). Also in the rotation, 6DOF Hands - SIVE was easier than Handle Bar - IVE ($Z = -2.097, p = 0.036$). To translate, 6DOF Hands - IVE was easier than 6DOF Hands - SIVE and Handle Bar - SIVE ($Z = -3.116, p = 0.002$ and $Z = -2.799, p = 0.005$). Scaling was easier with 6DOF - IVE and Handle Bar - IVE than with Handle Bar

- SIVE ($Z = -2.435$, $p = 0.015$ and $Z = -2.516$, $p = 0.012$). The preference of the techniques on the immersive system, specially 6DOF Hands, is due to its simplicity and ease of observation with Oculus Rift, as well as a bigger precision. On the interactive surface the objects could disappear from the screen with the side movements of the user, as the system tried to adapt the user's perspective of the objects according to his head's position.

The users preferred the immersive techniques, the favorite being 6DOF Hands - IVE, against Handle Bar - IVE, 6DOF - SIVE and Handle Bar - SIVE ($Z = -3.036$, $p = 0.002$ and $Z = -3.294$, $p = 0.001$ and $Z = -3.587$, $p < 0.001$). They also preferred Handle Bar - IVE against Handle Bar - SIVE ($Z = -2.353$, $p = 0.019$). The fun factor got similar results, with the users having more fun in 6DOF Hands - IVE, than in Handle Bar - IVE, 6DOF - SIVE and Handle Bar - SIVE ($Z = -3.082$, $p = 0.002$ and $Z = -3.800$, $p < 0.001$ and $Z = -3.970$, $p < 0.001$), and in Handle Bar - IVE than in Handle Bar - SIVE ($Z = -3.080$, $p = 0.002$). These preferences are in the same line as the interaction preferences.

Finally, comparing the systems, immersive and semi-immersive, the users felt more present in the immersive system, with that feeling being smaller in the semi-immersive system ($Z = -3.775$, $p < 0.001$). It is possible to observe each object from every different angle immersively, whereas semi-immersively the television's edges limited the visualization. To most of the users (seventeen), being able to see their arms on the immersive system helped them feeling more present on the virtual environment.

5 CONCLUSIONS

The constant technological advances have allowed the creation of more and more scenarios and environments that offer unique interaction capabilities. 3D objects are used in many different fields, such as assembly lines, health and architecture software, among others. However, the used methods to interact with 3D objects (mouse and keyboard) do not allow to explore all of the objects' functionalities.

So, in this work we tried to identify natural interaction techniques, which had to use gestures familiar to the users. After a study about existing techniques and works about manipulation on the tridimensional space, two techniques were chosen. These were implemented in two different environments, one immersive (IVE) and the other semi-immersive (SIVE). The chosen techniques were 6DOF Hands and Handle Bar.

To evaluate each technique in each environment, we carried out user test sessions. The users were asked to test each technique-system pair by performing three tasks of incremental difficulty. In each task the user had to fit a plug in a socket. The 6DOF Hands technique had higher precision and smaller task completion time, specially when rotations are needed. The completion time was also smaller in the IVE system. However the results of both techniques can still be improved, due to the lack of millimetric preci-

sion, essential in different fields.

The users felt it was easier to interact (move, rotate, scale) using the 6DOF Hands technique on the IVE system. This was also the users' preferred technique, being also regarded as the most fun to use. The techniques on the IVE system were considered easier to use, since on the SIVE system it was more difficult to view the objects.

Still, we believe it is possible to improve and complete this work. The implemented techniques for 3D object manipulation can be improved and there's also room to study and develop new interaction techniques, that can achieve greater precision than the tested ones, trying however to maintain the same level of familiarity to the users. It would also be relevant to try to combine new technologies that could allow to obtain the user's hands' orientation, without resorting to holding objects on their hands. With the gathered knowledge and conclusions obtained in this work, we created a foundation to support future works on this topic.

Acknowledgements

The work described in this paper was partially supported by the Portuguese Foundation for Science and Technology (FCT) through the project TECTON-3D (PTDC/EEI-SII/3154/2012), doctoral grant SFRH/BD/91372/2012 and by national funds through FCT with reference UID/CEC/50021/2013.

References

- [1] Aurélie Cohé, Fabrice Dècle, and Martin Hachet. tbox: A 3d transformation widget designed for touchscreens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2011.
- [2] Bruno R. De Araùjo, Géry Casiez, and Joaquim A. Jorge. Mockup builder: Direct 3d modeling on and above the surface in a continuous interaction space. In *Proceedings of Graphics Interface 2012*, 2012.
- [3] Otmar Hilliges, David Kim, Shahram Izadi, Malte Weiss, and Andrew Wilson. Holodesk: Direct 3d interactions with a situated see-through display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012.
- [4] Daniel Mendes, Fernando Fonseca, Bruno Araùjo, Alfredo Ferreira, and Joaquim Jorge. Mid-air interactions above stereoscopic interactive tables. 2014.
- [5] Peng Song, Wooi Boon Goh, William Hutama, Chi-Wing Fu, and Xiaopei Liu. A handle bar metaphor for virtual object manipulation with mid-air interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012.
- [6] Robert Wang, Sylvain Paris, and Jovan Popović. 6d hands: Markerless hand-tracking for computer aided design. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, 2011.



Computer Vision & 3D

Contents

Reconstrução de Superfícies Trianguladas a partir de Nuvens de Pontos sem Restrições Angulares <i>Gonçalo Leitão, Abel Gomes</i>	110
A GPU Approach to Augmented Reality using Non-Central Catadioptric Cameras <i>Tiago Dias, Pedro Miraldo, Nuno Gonçalves</i>	117
Especificação de Interrogações para Recuperação 3D em Ambiente Imersivo <i>Pedro Pascoal, Daniel Mendes, Alfredo Ferreira, Miguel Sales Dias</i>	125
Malhas Segmentadas com Resolução Variável <i>Rui S. V. Rodrigues, José F. M. Morgado, Abel Gomes</i>	133
Uma aplicação móvel para implementar o conceito de caderneta de cromos aos Museus <i>Luís Costa, António Coelho</i>	141

Reconstrução de Superfícies Trianguladas a partir de Nuvens de Pontos sem Restrições Angulares

Gonçalo Leitão
ESTG/IPG
Guarda
gleitao@ipg.pt

Abel Gomes
Instituto de Telecomunicações
Universidade da Beira Interior
agomes@di.ubi.pt

Sumário

A reconstrução de uma superfície triangulada a partir de uma nuvem de pontos é um problema difícil de resolver dado depender da forma do objeto original e da densidade dos pontos obtidos pelo dispositivo de aquisição 3D utilizado (scanner). Em grande medida, os algoritmos baseados em expansão da malha que existem na literatura não são bem-sucedidos porque os triângulos são apenas à malha em expansão com base em intervalos angulares que não podem ser ultrapassados. Por contraposição, propõe-se neste artigo um algoritmo que resolve este problema através da conjunção de funções que quantificam três propriedades geométricas fundamentais: coplanaridade, proximidade e regularidade.

Palavras-chave

Surface reconstruction, 3D point cloud, mesh growing.

1. INTRODUÇÃO

A reconstrução de superfícies a partir de uma nuvem de pontos é um importante tópico de investigação no domínio da computação gráfica. A sua aplicação em ciências e engenharias é variada; por exemplo, na modelação geométrica, realidade virtual, animação por computador, medicina ou engenharia reversa.

Na literatura encontramos vários métodos para a reconstrução de superfícies, os quais podem ser agrupados em três classes principais: *simpliciais* (ou trianguladas), *paramétricas* e *implícitas*. Os métodos simpliciais podem ainda ser subdivididos em dois grupos: os que são baseados em técnicas de Voronoi/Delaunay e os métodos de proliferação da malha (*mesh growing*). Neste artigo, só estamos interessados em métodos de reconstrução de superfícies trianguladas que seguem uma estratégia de proliferação da malha. Os métodos baseados na proliferação da malha geram uma superfície triangulada a partir de um triângulo inicial, a qual se vai propagando através do apensamento de novos triângulos, os quais terão de satisfazer um ou mais critérios geométricos.

Um desses algoritmos, que é baseado na proliferação da malha, é designado por *ball pivoting algorithm* (BPA) e foi introduzido por Bernardini et al. [Bernardini99]. Este algoritmo utiliza uma esfera assente em cada aresta fronteira da malha, e procura o próximo ponto da nuvem que se situa sobre a esfera de modo que mais nenhum outro ponto se situe dentro da esfera. Os três pontos sobre a esfera formam um possível triângulo, mas para isso é preciso que o ângulo diedral entre o triângulo da fronteira e o novo triângulo seja menor que 90° . Este algoritmo não é robusto nos casos em que a nuvem de pontos tem

densidade variável, tem ruído ou quando apresenta vincos (*ridges*); estes vincos correspondem a arestas no objeto original.

Li et al. [Li09] propõem um algoritmo de proliferação da malha que utiliza uma função de prioridade, que é dada pela soma conjunta de três grandezas. A primeira é o valor do cosseno do maior ângulo interno do novo triângulo, a segunda grandeza é o cosseno do ângulo diedral entre o triângulo da fronteira e o novo triângulo, e a terceira corresponde à distância da aresta fronteira ao ponto em avaliação. Nesta última grandeza, notar que a distância é normalizada relativamente ao raio de pesquisa usado para encontrar o conjunto dos pontos próximos. A função de prioridade aplica-se aos pontos candidatos para definir uma fila de prioridades de acordo com os respetivos valores. O ponto selecionado será o que tiver a maior prioridade. O algoritmo mostra insuficiências com modelos de formas complexas e impõe restrições às amplitudes dos ângulos internos de qualquer novo triângulo.

Angelo et al. [Angelo11] apresentam um algoritmo para a rápida reconstrução da superfície que se baseia no critério G2S, que é uma aproximação ao critério de Delaunay em 2D. No entanto, o algoritmo não apresenta bons resultados nos casos em que as nuvens de pontos não são uniformes.

O algoritmo proposto por Wongwaen et al. [Wongwaen12] subdivide o cubo que envolve a nuvem de pontos em pequenos cubos do mesmo tamanho, ou seja, de forma regular. De seguida, a triangulação dos pontos efetua-se no interior de cada cubo, sendo depois estas sub-triangulações ou sub-malhas posteriormente interligadas por triangulação intersticial. A progressão

dos triângulos é orientada pelo vetor de proliferação de cada aresta fronteira. Este vetor é perpendicular à aresta fronteira da malha em expansão e é coplanar com o triângulo fronteiro delimitado pela referida aresta, tal como se ilustra na Fig. 1.

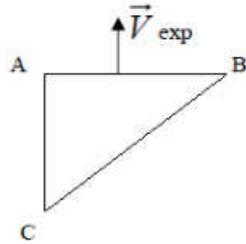


Figura 1: Vetor de proliferação da aresta AB.

Contudo, a proliferação da malha é condicionada pela imposição de limites às amplitudes dos ângulos internos do novo triângulo que irá ser anexado à malha. Além disso, a divisão uniforme do espaço circundante da nuvem de pontos pode originar triangulação intersticial errada no interior de um cubo. Por vezes, são unidos segmentos não adjacentes da superfície, pela razão de pertencerem ao interior do mesmo cubo em resultado do objecto apresentar uma variação rápida de curvatura. A partição uniforme do cubo que envolve a nuvem de pontos pode ainda ser inadequada quando a densidade dos pontos não é uniforme, podendo criar furos inexistentes no objeto original.

De modo semelhante, em [Wang13], e de modo a evitar transições drásticas na expansão dos triângulos, é limitada a variação angular relativamente à direção de proliferação. Assim, são colocadas restrições ao ângulo entre a normal do triângulo na fronteira da malha e a normal do novo triângulo. Além disso, e para evitar triângulos demasiado estreitos, são impostos limites às amplitudes dos ângulos internos dos novos triângulos.

Em [Xumin14], com o objetivo de reduzir o tempo e a complexidade, é proposto um algoritmo de reconstrução também baseado na expansão da malha e que utiliza a subdivisão *octree*, sem necessidade de processar o vetor normal em cada ponto. Contudo, o método de reconstrução da superfície também determina restrições à amplitude do ângulo diedral entre o novo triângulo e o triângulo de fronteira da malha, o mesmo acontecendo com os ângulos internos do novo triângulo.

Portanto, em geral, os problemas decorrentes da reconstrução de uma malha triangular relacionam-se com a dificuldade em encontrar critérios adequados que sustentem a construção de triângulos, em conformidade com a forma geométrica da superfície do objeto original. Por exemplo, regiões com curvatura acentuada ou vincada, bem como regiões de forma oscilatória, geram dificuldades na sua triangulação. A triangulação direta de uma nuvem de pontos também pode ser dificultada pela densidade variável (ou não-uniforme) dos seus pontos, em particular quando o objeto original possui furos. Além disso, não existe na literatura qualquer algoritmo baseado na proliferação da malha que construa uma

malha de triângulos tendencialmente regular, ou seja, triângulos tendencialmente equiláteros, a não ser que se use um passo de regularização dos triângulos após a construção da malha.

O algoritmo aqui apresentado, para reconstruir uma superfície triangulada a partir de uma nuvem de pontos, enquadra-se nos métodos de proliferação da malha. De modo diferente, este algoritmo utiliza funções que quantificam propriedades geométricas, tais como a *coplanaridade*, a *proximidade* e a *regularidade*, que são a base para a escolha do próximo triângulo a ser anexado à malha em crescimento. Com a aplicação de tais funções não existe a necessidade de estimar os vetores normais em cada ponto, e evita-se ainda a imposição de restrições angulares, quer à amplitude dos ângulos internos dos triângulos da malha, quer ao ângulo diedral entre triângulos adjacentes. A proliferação da malha vai avançando sempre pelas zonas onde se verifica menor variação de forma, ou seja, onde se verificam condições de maior coplanaridade. Deste modo, as regiões de maior curvatura ou afiadas são reconstruídas seguindo um caminho ou direção mais suave ou mais planar.

2. O ALGORITMO

O algoritmo consiste nos seguintes passos principais: (i) cálculo dos pontos vizinhos de cada ponto; (ii) construção do triângulo inicial; (iii) proliferação incremental da malha até à reconstrução completa da superfície. A descrição pormenorizada de cada um dos destes passos faz-se nas subsecções que se seguem. Mas antes disso, vejamos os conceitos fundamentais que sustentam o algoritmo aqui descrito.

2.1 Critérios geométricos

Como se referiu acima, há três critérios essenciais que são sintetizados no nosso algoritmo: proximidade, coplanaridade e regularidade.

A *proximidade* tem que ver com o cálculo dos pontos mais próximos e em redor de cada ponto. No algoritmo, iremos também calcular o ponto mais próximo de uma dada aresta. Assim, considerando os dois pontos extremos de uma aresta na fronteira da frente de expansão da malha, diz-se que um terceiro ponto é o ponto mais próximo a essa aresta, se for aquele que tem a menor soma das distâncias aos dois pontos extremos da aresta.

A *coplanaridade* é a propriedade dominante utilizada aquando da escolha do próximo triângulo a ser acrescentado à malha. Para isso, utiliza-se o conceito de vetor de proliferação associado a cada aresta da frente de expansão. Trata-se de um vetor coplanar com o respetivo triângulo da frente de expansão, que é perpendicular à própria aresta e é orientado para o seu exterior (Fig. 1). Este vetor é calculado tal como foi descrito em [Wongwaen12]. Note-se que o vetor de proliferação gera um semiespaço positivo que tem como fronteira o plano perpendicular ao triângulo e contém a aresta considerada na frente de expansão.

A *regularidade* da malha tem que ver com a produção de uma malha de triângulos que são tendencial e desejavelmente equiláteros. A quantificação da regularidade dos triângulos de uma malha faz-se do seguinte modo. Ao observarmos dois triângulos, aquele que geometricamente se aproxima mais de um triângulo equilátero é aquele onde a soma dos cossenos dos três ângulos internos tem maior valor. Este cálculo é efetuado pela aqui designada função de regularização (r).

2.2 Pontos Vizinhos

O primeiro passo do algoritmo tem como objetivo encontrar para cada ponto uma vizinhança de pontos próximos. Este conjunto de pontos vizinhos resulta da reunião de dois subconjuntos. Assim, dado um ponto A, o primeiro subconjunto de pontos vizinhos é formado pelos k ($=6$) pontos mais próximos de A. Note-se que o valor de 6 tem que ver com o facto de que numa malha regular cada ponto tem uma vizinhança de 6 triângulos (1 hexágono). Por sua vez, o segundo subconjunto de pontos vizinhos de A é constituído pelos pontos que incluem o ponto A como um dos seus k pontos mais próximos, mas que não pertencem ao primeiro subconjunto de pontos vizinhos de A. Nem sempre a relação de um ponto com os seus mais próximos é recíproca.

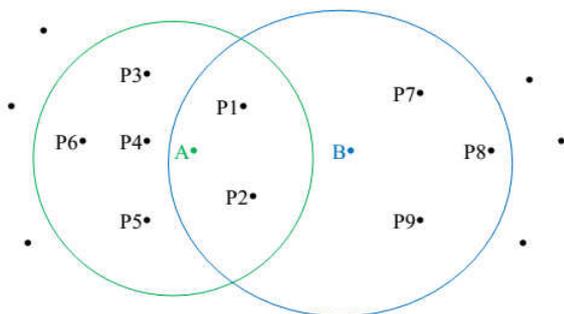


Figura 2: Pontos mais próximos de A e de B.

Por exemplo, na Fig. 2, o conjunto $\{P1, P2, P3, P4, P5, P6\}$ é o primeiro subconjunto da vizinhança de A, isto é, são os 6 pontos mais próximos de A. Quanto ao ponto B, o primeiro subconjunto da sua vizinhança é $\{P1, P2, P7, P8, P9, A\}$, que são os 6 pontos mais próximos de B. Embora o ponto B não seja um dos pontos mais próximos de A, acontece que o ponto A é um dos pontos mais próximos de B. Assim sendo, o ponto B é um dos elementos do segundo subconjunto da vizinhança do ponto A, ou seja, $V_A = \{P1, P2, P3, P4, P5, P6\} \cup \{\dots, B, \dots\}$. Notar que os pontos não irão ter as respetivas vizinhanças com igual dimensão. Deste modo, reduz-se o tempo de processamento em relação aos casos em que, *a priori*, se fixa o número de pontos vizinhos a considerar. Além disso, permite que possamos triangular zonas menos densas com outras mais densas.

Para diminuir o nível de complexidade e o tempo de procura para encontrar os pontos vizinhos, utiliza-se a

subdivisão *octree* da *bounding box* que contém a nuvem de pontos. A subdivisão é executada até que os nós folhas contenham um número de pontos inferior a um valor pré-definido, que no nosso caso é de 48 pontos. Note-se que este valor de 48 pontos foi escolhido tendo em conta que numa malha tendencialmente regular cada ponto tem uma vizinhança de 6 pontos, os quais por sua vez têm também uma vizinhança de 6 pontos, o que perfaz um total de 36 pontos, aos quais adicionámos mais 12 pontos por forma a não limitar demasiado o espaço de procura dos pontos. Deste modo, a procura dos pontos vizinhos fica circunscrita aos pontos que se encontram no respetivo nó folha e aos seus nós adjacentes.

Note-se ainda que a escolha deste valor de 48 pontos a utilizar como critério de paragem da subdivisão não tem qualquer influência sobre o resultado da reconstrução da superfície, embora deva ser um mínimo de 36 pontos pela razão referida acima. O objetivo é reduzir, para cada ponto, o domínio de procura dos pontos mais próximos, em vez de ser a totalidade dos pontos da nuvem. Refira-se também que quando consideramos o conjunto de pontos vizinhos de cada ponto, tal como foi descrito anteriormente, e unimos cada ponto a todos os seus vizinhos, gera-se um esqueleto unidimensional que se aproxima bastante do esqueleto final da malha que representa a superfície, como se ilustra na Fig.3.

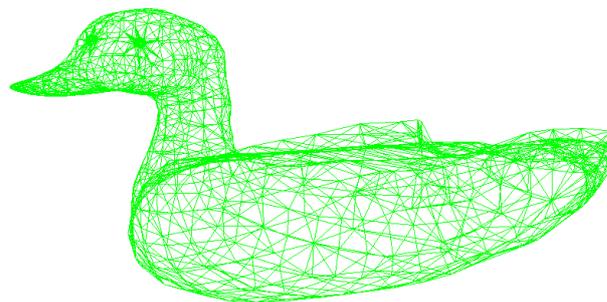


Figura 3: Esqueleto unidimensional gerado por proximidade.

2.3 Triângulo Inicial

A proliferação da malha tem início a partir das arestas de fronteira do triângulo inicial, como se ilustra na Fig. 4. Este triângulo deve encontrar-se numa região com relativa suavidade. Além disso, o triângulo inicial não pode ser degenerado (i.e., um triângulo alongado que quase se parece com um segmento de reta). Deve também ter-se em conta que o triângulo deve ser tangencial à superfície curva imaginária que interpola os pontos, pois, caso contrário, a reconstrução da superfície ficará logo comprometida à partida.

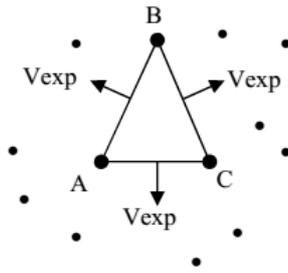


Figura 4: Triângulo inicial.

O triângulo é construído da seguinte forma. Dado um ponto e o seu vizinho mais próximo forma-se um segmento de reta. No conjunto dos pontos vizinhos de ambos os extremos desse segmento, encontrar o ponto mais próximo a esse segmento de reta. Estes três pontos formam um triângulo candidato a triângulo inicial. Se todos os restantes pontos vizinhos desses três pontos estiverem na união dos semiespaços definidos pelos vetores proliferação de cada aresta, então está encontrado o triângulo inicial (Fig. 4). Isto significa que, o triângulo inicial forma a secção de um prisma triangular onde todos os restantes pontos são exteriores a esse prisma.

2.4 Proliferação da Malha

Depois de construído o triângulo inicial, onde todas as arestas são fronteiras, tem início o processo de crescimento da malha. O princípio orientador do crescimento ou proliferação da malha é que esta avança sempre no sentido para onde se verificar maior coplanaridade, entre o plano do triângulo que suporta a aresta fronteira na frente de expansão e o plano do novo triângulo. Como consequência, a proliferação da malha vai progredindo preferencialmente pelas regiões de menor curvatura. Para cumprir este objetivo, em cada iteração do algoritmo são estudadas todas as arestas fronteiras, aplicando a cada uma delas um conjunto de critérios que identifica o triângulo candidato à anexação nessa aresta. Por fim, de entre os triângulos candidatos, o novo triângulo a ser construído será aquele que apresentar maior grau de coplanaridade com um dos triângulos da frente de expansão.

Note-se que, em [Wongwaen12] ou [Xumin14], a lista de arestas fronteiras é percorrida apenas até ser encontrada uma aresta onde o triângulo candidato verifica os critérios pré-definidos. Assim, inibe-se o conhecimento global da forma ao longo de toda a fronteira. Por outro lado, os critérios aplicados impõem limites aos valores dos ângulos internos do novo triângulo adjacente à aresta fronteira em estudo. No nosso caso, os critérios usados são funções que quantificam propriedades geométricas dos triângulos. A conjunção dos valores dessas funções é determinante na escolha do novo triângulo a ser construído em cada iteração.

O estudo aplicado a cada aresta fronteira para encontrar o triângulo candidato à anexação segue as seguintes condições:

- No conjunto dos pontos vizinhos do primeiro ponto extremo da aresta fronteira, encontrar o ponto mais próximo a essa aresta e que se localiza no semiespaço positivo definido pelo vetor de proliferação.
- Repetir o passo anterior para o segundo ponto extremo da aresta fronteira. Um destes dois pontos irá fechar o triângulo procurado e que será adjacente à aresta fronteira.
- Se os dois pontos obtidos coincidirem, então o triângulo candidato está encontrado.
- Caso os dois pontos encontrados não coincidam, então forma-se um quadrilátero, que poderá ser côncavo ou convexo, tal como se ilustra na Fig. 5.

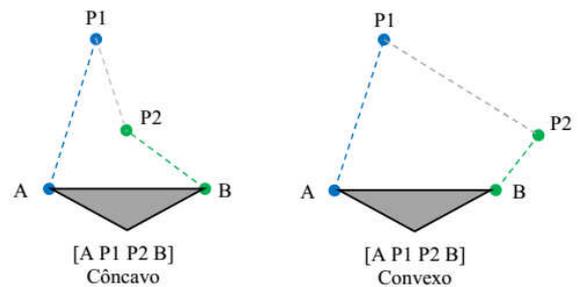


Figura 5: Classificação dos quadriláteros.

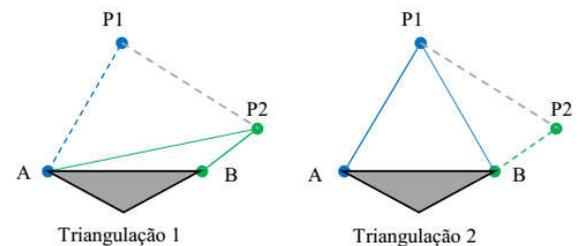


Figura 6: Triangulações possíveis do quadrilátero convexo.

- Caso o quadrilátero seja côncavo, o ponto mais próximo da aresta será o escolhido. Na Fig. 5, à esquerda, o ponto escolhido será o ponto P2.
- Caso o quadrilátero seja convexo são possíveis duas triangulações (Fig. 6). Neste caso, o ponto escolhido pode não ser o mais próximo da aresta, mas aquele que provocar uma subdivisão do quadrilátero de forma mais equilibrada.

Por exemplo, na Fig. 6, a triangulação que será escolhida será a segunda. Para selecionar a triangulação, usa-se a função soma dos cossenos dos três ângulos internos de um triângulo, já acima designada por função de regularização (r), e aplica-se a cada um dos triângulos resultantes da subdivisão do quadrilátero. O princípio seguido é o de maximizar a soma dos dois resultados e minimizar a sua diferença. Este princípio pode traduzir-se pela expressão a seguir:

$$r(A, P1, B) + r(B, P1, P2) - |r(A, P1, B) - r(B, P1, P2)|$$

No final de cada iteração, quando todas as arestas fronteiras da frente de expansão tiverem sido analisadas, o novo triângulo a construir será aquele que entre todos os triângulos candidatos resulte no maior grau de coplanaridade com o triângulo adjacente à aresta fronteira.

O novo triângulo dá origem a três casos quanto ao número de arestas novas a construir, como se ilustra na Fig. 7. Estes casos geram, respetivamente, 2, 1, ou 0 novas arestas. O primeiro caso, no qual se acrescenta duas arestas, acontece quando os vértices terminais da aresta da frente de expansão são ambos convexos. O segundo caso, em que se acrescenta uma aresta, surge quando um dos vértices da aresta da frente de expansão é côncavo e o outro é convexo. O terceiro caso dá-se quando ambos os vértices terminais são côncavos.

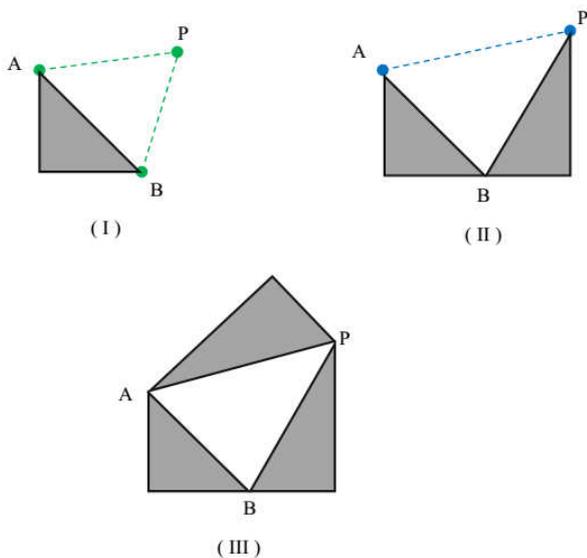


Figura 7: Topologias da reconstrução.

Obviamente, a reconstrução da malha termina quando já não houver arestas fronteiras.

3. RESULTADOS

3.1 Caracterização do ambiente de trabalho

O algoritmo proposto foi implementado em C++ com *OpenGL*, sobre *Microsoft Visual Studio 2010* em ambiente *Windows 8*. Os testes foram realizados em computador com processador *Pentium M* de *1.73 Ghz* e *2 GB* de *RAM*.

3.2 Modelos geométricos

Os modelos utilizados têm formato *obj* ou *pwn* e foram obtidos nos endereços, <http://tf3dm.com/3d-models/all> e http://home.eps.hw.ac.uk/~ab226/software/mpu_implicit/mpu_input_data.zip, e ainda no conjunto de modelos incluídos na aplicação *MeshLab v1.3.1*.

3.3 Avaliação qualitativa

A avaliação qualitativa do algoritmo pode ser efetuada com base nos problemas que foram sendo constatados na literatura, nomeadamente: características topológicas (e.g., furos), densidade não uniforme dos pontos, zonas vincadas, e zonas de curvatura acentuada ou com rápida variação de forma.

O critério da proximidade na computação localizada dos pontos mais próximos de cada ponto faz com que a reconstrução seja bem-sucedida mesmo em casos em que o objeto original tem características topológicas difíceis de reconstruir, como são o caso dos furos (Fig. 9(b) e Fig. 9(d)).

Note-se que a conjugação dos três critérios (i.e., proximidade, coplanaridade e regularidade) também resolve a questão da densidade não uniforme dos pontos, como é o caso da região do olho do pato representado na Fig. 8. Na verdade, pode observar-se que a região do olho tem uma densidade de pontos muito superior ao restante. Este caso mostra também que o algoritmo é bem sucedido em zonas vincadas. Repare-se que existe um vinco à volta do olho, algo que grande parte dos algoritmos não consegue reconstruir corretamente.

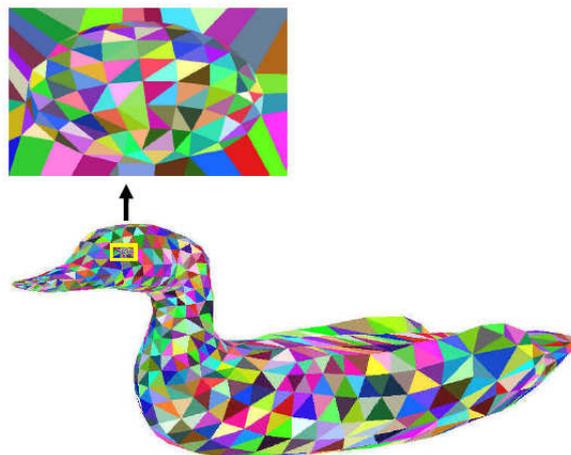


Figura 8: Malha com detalhe do olho.

As zonas vincadas são bem reconstruídas por causa da satisfação do critério da coplanaridade. Da mesma maneira, a prioridade dada ao critério da coplanaridade resolve o problema das zonas com curvatura acentuada ou com rápida variação da forma, como são os casos das barbatanas do golfinho na Fig. 9(a) ou o bico do pato na Fig. 9(c). No caso do golfinho, durante a reconstrução, a expansão da malha vai progredindo de ambos os lados da barbatana até às zonas de maior curvatura da barbatana. Só após as zonas mais complanares da barbatana estarem reconstruídas, é que o algoritmo avança para as zonas menos complanares (i.e., com mais curvatura). O mesmo se observa no caso do bico do pato.

Portanto, a coplanaridade é o critério chave que torna o algoritmo simples, eficiente, correto e robusto. Repare-se que, o algoritmo impõe que cada aresta só poderá ter no máximo dois triângulos incidentes, pelo que, logo à

partida está garantida a correção topológica da malha reconstruída.

Refira-se também que a regularidade dos triângulos faz com que a malha final tenha à partida um número menor de triângulos relativamente a outros algoritmos existentes na literatura. Por exemplo, o modelo da Fig. 9(d) foi

gerado com 2880 triângulos, o que acontece também com a aplicação *MeshLab*, mas o algoritmo proposto em [Xumin14] gera uma malha com 3000 triângulos para a mesma nuvem de pontos.

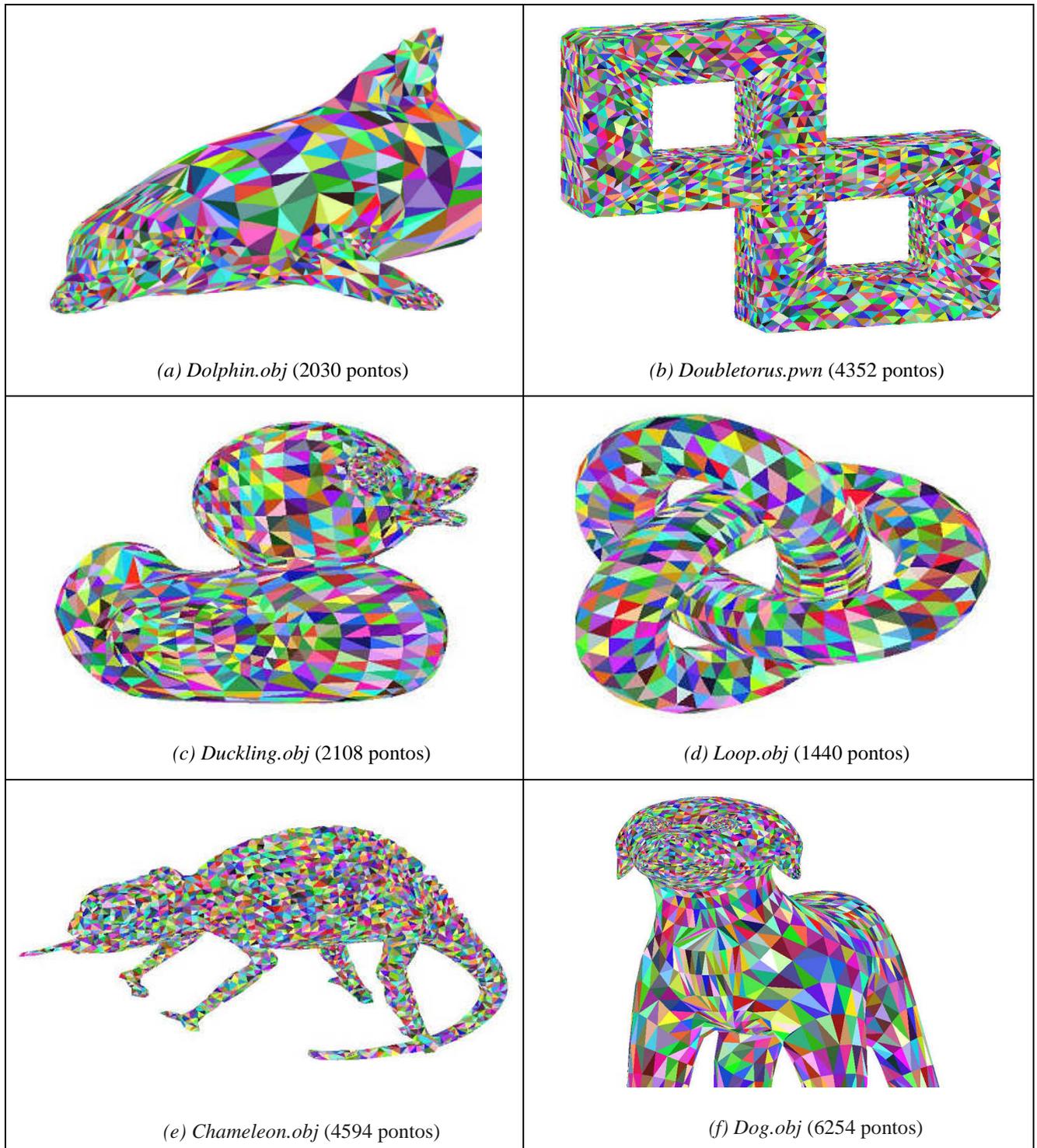


Figura 9: Malhas reconstruídas a partir de nuvens de pontos.

4. CONCLUSÃO

Foi proposto um algoritmo para reconstrução de malhas triangulares a partir de nuvens de pontos que se enquadra nos métodos de proliferação da malha (*mesh growing*). O método proposto resolve o problema das restrições angulares, quer dos ângulos internos dos triângulos quer da direção de expansão. O problema foi resolvido pela aplicação de funções que quantificam propriedades geométricas como a coplanaridade, regularidade e proximidade. Mais ainda, a reconstrução avança após o conhecimento global ao longo da fronteira e não apenas localmente. Os resultados mostram o bom funcionamento do algoritmo, mesmo em casos onde usualmente existem dificuldades devido à geometria ou topologia.

5. REFERÊNCIAS

- [Angelo11] L. D. Angelo, P. H. Stefano, L. Giaccari. A new mesh-growing algorithm for fast surface reconstruction. *Computer Aided Design*, Vol. 43, no. 6, 639-650, 2011.
- [Bernardini99] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, G. Taubin. The ball pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, no. 4, 349-359, 1999.
- [Li09] X. K. Li, C. Y. Han, W. G. Wee. On surface reconstruction: A priority driven approach. *Computer Aided Design*, Vol. 41, no. 9, 626-640, 2009.
- [Wang13] N.Wang, Q. Zhang, D. Zhou, X. Wei. Local optimum triangulation for unorganized point cloud. *Research Journal of Applied Sciences, Engineering and Technology*, Vol. 6, no.10, 1862-1867, 2013.
- [Wongwaen12] N.Wongwaen, S. Tiendee, C. Sinthanayothin. Method of 3D mesh reconstruction from point cloud using elementary vector and geometry analysis. *International Conference on Information Science and Digital Content Technology (ICIDT 2012)*, Vol. 1, 156-159, June 26-28, Jeju, South Korea, 2012.
- [Xumin14] Liu Wongwaen, Yang Lixin, Li Cailing. A robust mesh growing surface reconstruction algorithm based on octree. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, Vol. 7, No. 3, 135-146, 2014.

Este trabalho encontra-se ainda em desenvolvimento, motivo pelo qual não são apresentadas comparações quantitativas com outros métodos segundo alguma métrica. Este será um dos objetivos do trabalho futuro, assim como a otimização computacional de alguns procedimentos para tornar o algoritmo ainda mais eficiente e robusto, em particular em modelos de maior tamanho em termos do número de triângulos, bem como avaliar a sua aplicação com outros tipos de geometrias.

A GPU Approach to Augmented Reality using Non-Central Catadioptric Cameras

Tiago Dias
ISR
University of Coimbra
tdias@isr.uc.pt

Pedro Miraldo
ISR/IST
University of Lisbon
pmiraldo@isr.tecnico.ulisboa.pt

Nuno Gonçalves
ISR
University of Coimbra
nunogon@isr.uc.pt

Abstract

This paper addresses the problem of augmented reality on images acquired from non-central catadioptric systems. We propose a solution which allows us to project textured objects to images of these types of systems in real-time which, depending on the complexity of the objects, can run up to 20 fps with a reasonable resolution (near to a real-time framerate). The main contributions of our solution are related with the differences on the image formation: projection of the 3D segments onto the image of non-central catadioptric cameras; occlusions; and illumination/shading. To validate the proposed solution, we used a non-central catadioptric camera formed with a perspective camera and a spherical mirror. We used three distinct and well known objects in computer graphics: “bunny”, “happy buddha” and “dragon”, from Stanford database.

Keywords

Augmented Reality, Non-Central Catadioptric Cameras, Parallel Computing.

1 Introduction

Geometrically, any imaging device can be modeled by the association between image pixels and unconstrained 3D straight projection lines [Grossberg and Nayar, 2003]. When all cameras’ 3D projection lines intersect at a single 3D point (also called effective viewpoint), imaging devices are called central. On the contrary, when they do not intersect at a single point, camera systems are called non-central.

Most state-of-the-art computer vision and computer graphics methods/algorithms were developed under the assumption that images are acquired by sensors verifying the pinhole camera model (perspective camera [Hartley and Zisserman, 2000]), thus free from distortions. The main reason for the use of perspective central cameras is its simplicity (specially in what is related to its projection model). However, in the last few years, several new types of cameras with different capabilities have been developed. Often, the goal is to achieve 360-degree field of view (usually called omnidirectional cameras), which is very useful for applications ranging from panoramic photography to robotics and medical imaging.

With appropriate undistortion methods, any central camera system can be modeled by a central perspective camera, [Swaminathan et al., 2003]. As a result, the same methods/algorithms can be easily applied to all central camera systems. For these reasons, when possible, researchers tried to design new camera systems that verify the single

view point constraint, central cameras. The first central omnidirectional camera system was proposed by Nalwa in 1996 [Nalwa, 1996], which consists in aligning four perspective cameras with four mirrors. Later (following Nayar’s work [Nayar and Baker, 1997]), several authors started to build omnidirectional cameras combining perspective cameras with quadric mirrors (catadioptric camera systems). In theory, as shown in [Baker and Nayar, 1999], it is possible to define a set of condition (using specific types of mirrors and a perfectly alignment between the camera and mirror) which ensures that such systems are central. However, small misalignments (for example between the camera and mirror(s)) or using other types of mirrors (for examples spherical mirrors) will imply that these systems will not verify the single viewpoint constraints. This means that, in practice, all omnidirectional catadioptric systems are non-central cameras [Swaminathan et al., 2001]. For these non-central camera systems, distortion cannot be modeled without prior knowledge of the 3D world from the scene (unwrapped images cannot be recovered) [Swaminathan et al., 2003], which means that new methods/algorithms have to be developed.

Augmented reality has been studied for almost fifty years [Appel, 1968]. As stated by Azuma [Azuma, 1997], augmented reality can be defined as the projection of virtual 3D objects to the image plane. For central cameras, a large number of distinct methods have been presented, e.g. [Fournier et al., 1993, Sato et al., 1999,

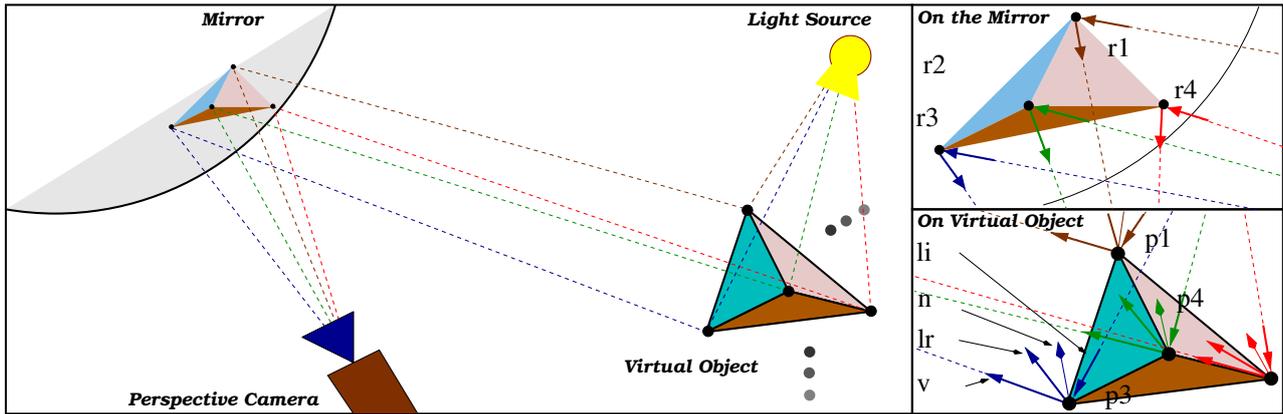


Figure 1: Representation of the projection and illumination steps. Since we are considering very small triangles, we assume that their illumination is constant – Flat shading technique [Hughes et al., 2014]. To represent the illumination in each triangle, we consider its mass center ${}^{(j)}\mathbf{t}$ and its reflection point on the mirror ${}^{(j)}\mathbf{r}_r$, for all j

Debevec, 2008, Santos et al., 2012]. However, to the best of our knowledge, augmented reality on images acquired by non-central catadioptric cameras was not yet addressed and, as explained before, these conventional approaches can not be directly applied.

Augmented reality, for these types of cameras, is extremely useful for human-computer interaction [Goodrich and Schultz, 2007], with several important applications in robotics. Two examples of these applications are: teleoperation [Chintamani et al., 2010] (creation and projection of 3D virtual landmarks to assist the human on robot navigation) and the creation of augmented reality environment simulations [Chen et al., 2009] (creation and projection of 3D objects to simulate real scenarios). Another example of an environment simulation (using augmented reality) is its application on the medical surgery (see e.g. [Fuchs et al., 2009]). To conclude, note that medical doctors are used to work directly with raw distorted imagery.

This paper aims at studying the effects of the non-central catadioptric image formation on an augmented reality framework. We went through all basics steps required for the use of augmented reality (camera calibration, object segmentation, projection, illumination/shading, and occlusions), analyzing which of them can be solved using existing methods and which require changes. The resulting framework can be divided into two stages (which will be denoted as pre-processing and real-time). Pre-processing stage will include all steps that can be computed *a priori* (avoid unnecessary computation steps that could increase the computation time) while the real-time stage include steps that depend on certain parameters such as camera and light source positions. The main contributions of this paper are:

- projection of the object’s skeleton, Sec. 2.2.1, that consists in the projection of the object’s skeleton to the non-central catadioptric image;
- occlusions, Sec. 2.2.2, one needs to verify if the

- pieces (already projected to the image) are overlapped and, if they are, verify which of them are visible; and
- illumination and shading, Sec. 2.2.3, that give shape to the projection of the 3D object.

2 Augmented Reality using Non-Central Catadioptric Cameras

Augmented reality must take into account the following steps: camera calibration (intrinsic and extrinsic parameters), 3D Object triangulation, skeleton projection (projection of the 3D triangles that define the object), occlusions, and illumination/shading. In this paper we are assuming that the virtual object is rigid and static. In the next subsections, we describe the contents of the framework stages.

2.1 Pre-Processing Stage

The pre-processing stage consists on a set of algorithms that, since they do not depend on illumination and camera localization parameters, can be computed *a priori*. Thus, on this stage we include the following three steps: camera calibration, 3D segmentation of the object, and its texturization. Camera calibration consists on the estimation of the parameters that map image pixels to 3D straight lines which, for non-central camera models, is not trivial as the calibration of perspective cameras. However, several authors proposed algorithms for the calibration of non-central catadioptric camera systems (e.g. [Micusik and Pajdla, 2004, Perdigoto and Araujo, 2013, Agrawal and Ramalingam, 2013]), which consists on the estimation of both mirror and camera parameters.

The second step of the pre-processing stage is related with the segmentation of the 3D virtual object. As described in the introduction, the virtual object must be decomposed into small 3D features to later be projected onto the image plane. We used the segmentation of the 3D virtual object in 3D triangles. To evaluate our method we used a virtual cube (which we had to triangulate) and well known objects from Stanford database

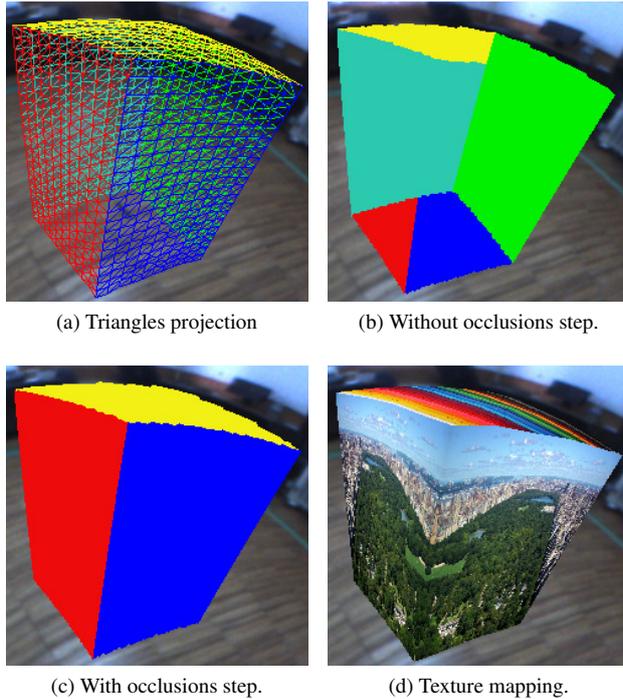


Figure 2: Results of the triangles' projection and occlusion steps applied to a 3D virtual cube. Fig. (a) shows the projection of the 3D triangles onto the image. Figs. (b) and (c) show the effects of the occlusion step (before and after respectively) and Fig. (d) shows the result of the occlusion step with textured faces. The effects of distortion can be easily seen from any of these images.

[Stanford University CG Lab, 1993]. From now on, we will assume that the camera is calibrated and that objects are triangulated and textured.

2.2 Real-time Stage

Real-time stage is related with the steps that must be computed each time a new frame is received. Thus, we include the following four steps: skeleton's projection, occlusions, illumination and display. All these steps depend on the geometry of the imaging device and, as we describe in the introduction, since for images of non-central camera models we cannot get unwrapped images, new algorithms have to be defined.

2.2.1 Projection

Assuming that we know the camera calibration and that our 3D object is triangulated and textured, one of the most challenge steps is the projection of these 3D triangles (which form the 3D objects) to the image plane. Considering that the triangles are small enough, the effects of distortion are neglectable [Swaminathan et al., 2003]. Using this assumption, to project these 3D triangles, we just need to take into account the projection of three 3D points to the image plane (vertices of the triangle). Contrary to the projection of 3D points to perspective cameras, this projection for non-central catadioptric camera systems is quite complex (e.g. [Gonçalves, 2010, Agrawal et al., 2011]). In

addition, one has to verify if the coordinate system of the virtual object is aligned with the camera's coordinate system (this is very important when a mobile camera is used). This problem is known as the absolute pose problem and there are several solutions in the literature (e.g. [Schweighofer and Pinz, 2008]). Let us consider the superscripts (\mathcal{W}) and (\mathcal{C}) to represent features in the world (in which the 3D virtual object was defined) and the camera coordinate system, respectively. Let us denote the vertices of the triangle as $\mathbf{p}^{(\mathcal{W})} \in \mathbb{R}^3$. The goal is to compute the rigid transformation $\mathbf{H}^{(\mathcal{C}\mathcal{W})} \in \mathbb{R}^{4 \times 4}$ that transforms points from the world to the camera coordinate systems such that

$$\tilde{\mathbf{p}}^{(\mathcal{C})} \sim \mathbf{H}^{(\mathcal{C}\mathcal{W})} \tilde{\mathbf{p}}^{(\mathcal{W})}, \quad (1)$$

where $\tilde{\mathbf{p}}$ denotes the homogeneous representation of \mathbf{p} . Each time a new image is received, the pose must be re-computed. From now on, we will assume that 3D points are already known in the camera coordinate system.

Let us now consider the projection of the three 3D points that define all the triangles' vertices. For all these points ${}^{(j)}\mathbf{p}$ (j^{th} vertex of the j^{th} triangle), one needs to compute their respective reflection points on the mirror ${}^{(j)}\mathbf{r}$ (see Fig. 1). To compute these reflection points we can use [Gonçalves, 2010, Agrawal et al., 2011]. Yet, these methods are quite complex and the goal of this paper is not to address this problem. Therefore, we will consider this as a black box. However, one have to take into account the computation effort required for this projection. Unlike the perspective case, where the projection of 3D points only requires a simple matrix multiplication which can be computed very fast (matrix of the camera's internal parameters times the 3D point), the computation of the precise reflection point requires much more computation effort. To get reasonable computation times, we had to implement these computations using the CUDA toolkit.

It can now be assumed that, for all the 3D points, we know its respective reflection point. Moreover, from the camera intrinsic parameters, one can write

$$\left\{ {}^{(j)}\mathbf{u}, {}^{(j)}\mathbf{u}, {}^{(j)}\mathbf{u} \right\}, \text{ where } {}^{(j)}\mathbf{u} \sim \mathbf{K}^{(j)} \mathbf{r} \text{ and } {}^{(j)}\mathbf{p} \mapsto {}^{(j)}\mathbf{r}, \quad \forall j = 1, \dots, N, \quad (2)$$

${}^{(j)}\mathbf{u}$ are the coordinates of the vertices on the image plane and $\mathbf{K} \in \mathbb{R}^{3 \times 3}$ are the camera intrinsic parameters [Hartley and Zisserman, 2000]. An example of the proposed solution is shown in Fig. 2(a).

Since we are ignoring the distortion and using projected coordinates of the three vertices of each triangles, the texture matching can be computed by using a simple affine transformation between the texture on the 3D triangle and its projection on the image. Fig. 2(d) shows these results. Note that this image already takes into account occlusions step, which will be analysed in the next subsection. From now on, we assume that, for each new image frame, we know the projection to the image of all 3D triangles that define the object.

$$\begin{aligned}
{}^{(j)}I^{(ch)} = & \overbrace{K_e^{(ch)} + G_a^{(ch)} K_a^{(ch)} + \text{spot}_k \sum_{k=1}^M \underbrace{{}^{(k)}L_a^{(ch)} K_a^{(ch)}}_{\text{ambient component}}}^{\tilde{I}^{(ch)}} + \\
& + \text{spot}_k f_k \text{occ}_k \sum_{k=1}^M \underbrace{\left(\underbrace{{}^{(k)}L_d^{(ch)} K_d^{(ch)} \max \left\{ -\binom{(j)}{(k)}{\mathbf{l}_i}^T \binom{(j)}{(k)}{\mathbf{n}_t}, 0 \right\}}_{\text{diffuse component}} + \underbrace{{}^{(k)}L_s^{(ch)} K_s^{(ch)} \max \left\{ \binom{(j)}{(k)}{\mathbf{v}_i}^T \binom{(j)}{(k)}{\mathbf{l}_r}, 0 \right\}^{sh}}_{\text{specular component}} \right)}_{\binom{(j)}{\tilde{I}_k}^{(ch)}}} \quad (3)
\end{aligned}$$

Illumination equation for a single 3D point using non-central catadioptric cameras: M is the number of light sources; $K_a^{(ch)}$, $K_d^{(ch)}$, $K_s^{(ch)}$, $K_e^{(ch)}$ and sh are ambient, diffuse, specular, emission, shininess material color properties; $G_a^{(ch)}$ is the global ambient light property ((ch) denotes the color channel); ${}^{(k)}L_a^{(ch)}$, ${}^{(k)}L_d^{(ch)}$, ${}^{(k)}L_s^{(ch)}$ are the ambient, diffuse and specular properties of the k^{th} light source; boolean parameters f_k and occ_k are used to control whether the point is illuminated or not; and spot_k controls the cutoff angle of the light source (definition of spotlight). A graphical representation of the directions $\binom{(j)}{(k)}{\mathbf{l}_i}$, $\binom{(j)}{(k)}{\mathbf{l}_r}$, $\binom{(j)}{(k)}{\mathbf{n}_t}$, and $\binom{(j)}{(k)}{\mathbf{v}_i}$ is shown in Fig. 1.

Algorithm 1: Reformulation of painter’s algorithm for images of non-central catadioptric cameras.

Let $\binom{(j)}{(i)}{\mathbf{p}}$ be the 3D coordinates of the i^{th} vertex of the j^{th} triangle and N the number of existing triangles:

for $j = 1$ to N **do**

Compute mass center $\binom{(j)}{(1)}{\mathbf{t}}$ for each triangle $\left\{ \binom{(j)}{(1)}{\mathbf{p}}, \binom{(j)}{(2)}{\mathbf{p}}, \binom{(j)}{(3)}{\mathbf{p}} \right\}$;
 Compute reflection point $\binom{(j)}{(1)}{\mathbf{r}_t}$,
 using [Gonçalves, 2010, Agrawal et al., 2011];
 Set $\binom{(j)}{(1)}{\xi}$ as the distance between $\binom{(j)}{(1)}{\mathbf{r}$ and $\binom{(j)}{(1)}{\mathbf{t}}$;

end

Sort all the triangles by descendant order using the computed $\binom{(j)}{(1)}{\xi}$, for all $j = 1, \dots, N$;

2.2.2 Occlusions

Now, let us consider the occlusions’ problem (very well known problem in 3D computer graphics). For all the projected segments, it is fundamental to understand if they are overlapped and, if they are, which of them are in front. To solve this problem for images of perspective cameras, several solutions were proposed, for example: the Painter’s algorithm [Hughes et al., 2014], Z-Buffer (also known as Depth Buffer) [Hughes et al., 2014] and A-Buffer [Carpenter, 1984].

Z-Buffer is probably the simplest and most widely used technique to solve this problem. However, this method requires the association between pixels and coordinates of 3D points, for all pixels that define the object. We want to avoid this because of the complexity associated with the projection of points using non-central catadioptric systems (described in the previous section). Moreover, as described in the previous section, we are ignoring the distortion effects on the projection of the triangles (by considering a large number of small 3D triangles), which means that there is no easy way to compute the matching between all pixels and respective 3D points that belong to the triangles.

Our goal is just compute those triangles that are in front and make sure that they are visible. Then, we propose a simple solution based on painter’s methodology. Since we are using non-central catadioptric imaging systems, con-

ventional algorithms cannot be used. These methods need to be reformulated taking into account the geometry of the imaging device. The goal of painter’s methodology is to organize 3D triangles as a function of the distance of these triangles to the camera system. Then, the problem is solved by displaying the 2D triangles using this order. The main difference between the proposed method and conventional painter’s algorithm is related with the definition of “point of view”, required for the computation of the distance between the image device and 3D points. If for central cameras one can use the camera center (also called the effective view point [Hartley and Zisserman, 2000]) as the referential for the distance, in our case this cannot be applied (non-central catadioptric system). Thus, to compute the distance between the 3D triangles and the camera system we consider the distance between the triangle (since the triangles are very small, we used the mass center of the triangle) and the respective 3D reflection point on the mirror (see Fig. 1). The proposed solution is formalized in Algorithm 1. After the application of this algorithm, we have the 2D triangles in descending order and ready to display. The application of this step can be seen by Fig. 2(b), without applying the proposed algorithm, and Fig. 2(c), after the application of the proposed Algorithm 1.

2.2.3 Illumination

When considering a 3D object with a solid color, without illumination, the projection of this 3D object to the image will be represented by a BLOB (Binary Large Object), see Fig. 3(a). The use of an illumination model and a shading technique will create the illusion of shape to the projection of the 3D object. For the perspective camera, to compute the intensity of light (illumination) associated to a single pixel, two different models were proposed: Phong reflection model and Torrance-Sparrow reflection model (both described at [Blinn, 1977]). Also, to solve the shading problem, several techniques were proposed, such as: Flat shading [Hughes et al., 2014], Gouraud shading [Gouraud, 1971], and Phong shading [Phong, 1975]. We want to stress out that this step depends on the geometry of the imaging device, which means that these conven-

tional techniques cannot be used directly.

We decided to derive a very simple algorithm. Illumination model has to be reformulated. We start from the Phong's reflection model [Blinn, 1977] and derive a solution to work with non-central catadioptric cameras, Eq. 3. The three color channels are computed separately. For each channel and for a single point (on the image), there is two illumination components: $\tilde{I}^{(ch)}$, which represents the influence of both global and light source ambient properties on the object's material; and $\check{I}^{(ch)}$ which represents the influence of the diffuse and specular light source properties on the object's material. The first one does not depend on the geometry of the camera systems and does not require further analysis. On the other hand, the latter depends on the object's projection to the image. Next we analyse in more detail the diffuse and specular components. The illumination on a single 3D point on the object must include the following components:

- **Diffuse reflection:** component that defines the object shape. It depends on the direction of the incident ray (that comes from the light source) and the surface normal at the respective 3D point (vertex position). A graphical representation can be seen at Fig. 1;
- **Specular reflection:** is associated with the shininess reflected by the object. It depends on the reflection ray (that comes from light source) and direction to the viewer's position. The incident ray is known (which is given by the position of the light source) and we can obtain its reflection ray using the Snell's law¹. Since we are using non-central system, the direction to viewer's position can not be computed using conventional techniques. For central cameras, this direction is computed by considering viewer's position at the "single view point" (3D point where all 3D projection lines intersect) which, as we already described, in our case does not exist. Thus, to solve this problem, we define the viewer's position at the respective reflection point on the mirror, which can be computed using [Gonçalves, 2010, Agrawal et al., 2011]. This relationship is depicted at Fig. 1.

This components are computed for all the vertices of the triangles.

Regarding the shading, we could use the variations of Flat, Phong, or Gouraud's methodologies. To test the robustness of our framework, we used both Flat and Gouraud approaches. Based on the experimental results, as expected, we verified that Gouraud's method gives the best results. Note that Gouraud's technique (as the shading method) allows a smoother transition between the triangles. The proposed solution is formalized in Algorithm 3.

Let us now consider the case where a triangle is covered by another one, in relation to the spotlight. In this case, the triangle should not be illuminated. However, the proposed

¹The Snell's law is given by $\mathbf{l}_r = \mathbf{l}_i - 2(\mathbf{l}_i^T \mathbf{n}_i) \mathbf{n}_i$, where \mathbf{l}_i , \mathbf{l}_r and \mathbf{n}_i are the incident ray, reflected ray, and the surface normal at the respective 3D point.

Algorithm 2: Sort and verify occlusions on the projected triangles.

```

for  $j = 1$  to  $N$  do
    Compute mass center  ${}^{(j)}\mathbf{t}$ ;
    Set  ${}^{(j)}\xi$  as the distance between  $\mathbf{s}$  and  ${}^{(j)}\mathbf{t}$ ;
    for  $l = j + 1$  to  $N$  do
        Compute piramide  $\Omega$ ;
        Considering each  ${}^{(l)}\mathbf{p}$  vertices of the  $l^{\text{th}}$  triangle;
        if  ${}^{(l)}\mathbf{p}$  is inside  $\Omega$  then
            Set  ${}^{(l)}occ_l = 0$ ;
        end
    end
    Compute new  ${}^{(j)}I^{(ch)}$  for the triangle;
end
Sort all the triangles by ascendant order using  ${}^{(j)}\xi$ ;
    
```

Algorithm 3: Proposed illumination algorithm.

Let ${}^{(j)}\mathbf{p}$ be the 3D coordinates of the i^{th} vertex of the j^{th} triangle, N the number of existing triangles. M the number of light sources, $\mathbf{d}_{sl(k)}$ the direction of the spotlight and Ω the union between the spotlight and j^{th} triangle's edges:

```

for  $j = 1$  to  $N$  do
    Compute vertices' normal  ${}^{(j)}\mathbf{n}_i$ ;
    Compute the reflection points  ${}^{(j)}\mathbf{t} \mapsto {}^{(j)}\mathbf{r}_i$ ;
    Compute the visualization vectors  ${}^{(j)}\mathbf{v}_i$ ;
    Set  ${}^{(j)}I^{(ch)} = {}^{(j)}\check{I}^{(ch)}$  for each vertex;
    for  $k = 1$  to  $M$  do
        Compute the reflection rays  ${}^{(j)}\mathbf{l}_{r(k)}$ ;
        Set  ${}^{(j)}f_k = 1$  and  ${}^{(j)}spot_k = 0$ ;
        if angle between  ${}^{(j)}\mathbf{l}_{r(k)}$  and  ${}^{(j)}\mathbf{n}_i$  bigger than zero then
             ${}^{(j)}f_k = 0$ ;
        end
        if maximum of  $\langle {}^{(j)}\mathbf{l}_{r(k)}, \mathbf{d}_{sl(k)} \rangle$  and 0 bigger than  $C^{te}{}^{(k)}$  then
             ${}^{(j)}spot_k = \max\{ {}^{(j)}\mathbf{l}_{r(k)}^T \mathbf{d}_{sl(k)}, 0 \}^{\epsilon}$ ;
        end
        Add  ${}^{(j)}I^{(ch)} = {}^{(j)}I^{(ch)} + {}^{(j)}\check{I}_k^{(ch)}$  for each vertex, see Eq. (3);
    end
    Calculate  ${}^{(j)}I^{(ch)}$  using a linear interpolation of  ${}^{(j)}I^{(ch)}$ ;
end
    
```

Algorithm 3 does not solve this problem. This does not depend on the geometry of the imaging device and there are several solutions in the literature that can be used to solve this problem. We used a simple solution which, basically, search if a point k is occluded and, if it is, sets $occ_k = 0$ (otherwise it will be $occ_k = 1$). This variable will be used as parameter of Eq. 3. Since we used the Gouraud's methodology, to compute the triangle's illumination, we just need to make a linear interpolation between the colors of the three vertices (that define each triangle). Results obtained using our framework, with and without the proposed illumination technique, can be seen at Figs. 3(a) and 3(b).

3 Experimental Results

We used a non-central catadioptric camera formed with a perspective camera and a spherical mirror. We have implemented the proposed steps using C/C++ pro-

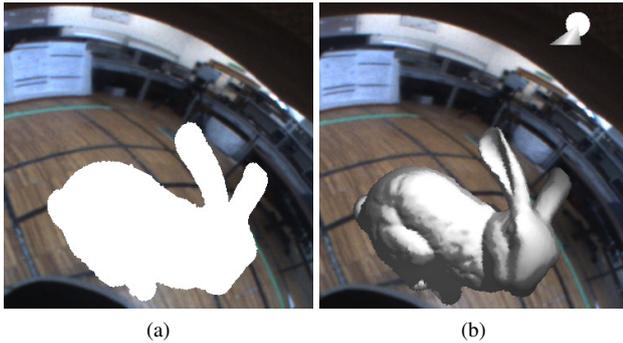


Figure 3: Results of the application of the illumination step to the “bunny” object. In Fig. (a) we show the results without the illumination step. In Fig. (b) we show the same results with the illumination step.

gramming and the CUDA toolkit (from NVIDIA). The framework was tested on a computer with: “Intel i7 3630QM” CPU (2.4 GHz with 4 cores); and “NVIDIA GeForce GT 740M” GPU (810 MHz with 384 CUDA cores). We used three 3D virtual objects: “bunny”, “happy buddha”, and “dragon” objects, from Stanford database [Stanford University CG Lab, 1993]. In addition, results with the triangles’ projection and occlusions steps are shown at Fig. 2. Results showing the advantages on the use of illumination can be seen in Fig. 3.

To evaluate the complete framework, we use a moving light source. The results are shown in Fig. 5 (a video with the complete sequence is sent in the supplementary material). In addition, we repeated these tests using different number of triangles that define each objects (approximately 300 times each, at different spotlight positions), storing the respective computation times. The results are shown in Fig. 4(b). In addition, to further evaluate the computation complexity of the proposed solution, we consider the virtual cube, varying the number of triangles that define the cube. The results are shown in Fig. 4(a). As expected, the execution time is higher (inverse of the frames per second) when the number of triangles (that form the 3D object) increases. By the analysis of both graphics of Fig. 4, one can observe that the increase of number of triangles leads us to a less number of outliers, and consequently to a more stable number of frames per second.

In addition, we also considered an experiment with the camera at different position and orientations. The camera location is computed in real-time and the proposed framework is applied to the system. The results are shown in Fig. 6 (a video with the complete sequence is sent in the supplementary material).

For all experiments, the virtual 3D object is positioned at the center of the arena, with the spotlight pointing to the object’s position. For the “bunny”, “buddha” and “dragon” we used white, gold and red colors for the light sources, respectively. For all the objects, it was used a silver color as the material property ($K_a^{(ch)}$, $K_d^{(ch)}$, $K_s^{(ch)}$ parameters pre-

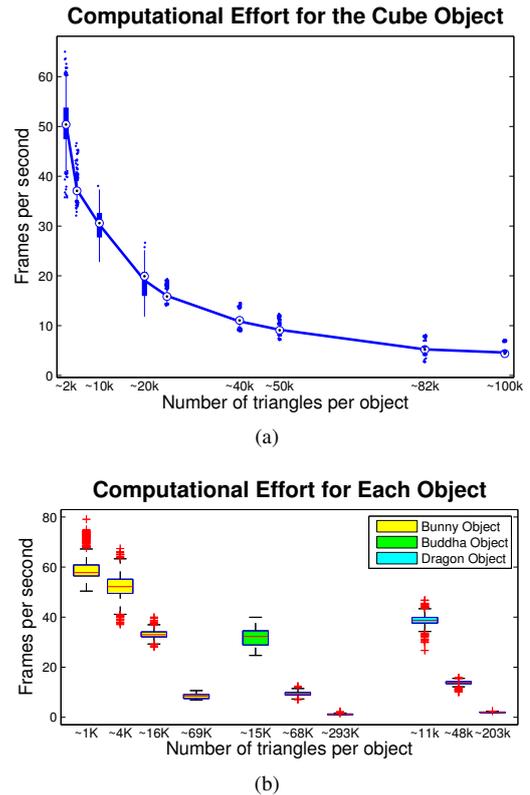


Figure 4: Results of the computation effort for all the 3D objects. In Fig.(a) we present the number of frames per second obtained using different number of triangles for the cube object. In Fig.(b) we present the relation between the number of frames per second and the number of triangles for the “bunny”, “buddha” and “dragon” objects.

sented in Eq. 3).

To conclude, we considered an experiment with a moving camera. The camera location is computed in real-time and the proposed framework is applied to the system. The results are shown in Fig. 6. A video with the complete sequence is sent in the supplementary material.

4. Conclusions

In this paper we address the use of Augmented Reality on images of a non-central catadioptric system. We believe that this is the first time that this problem is addressed. The goal of this paper is to identify differences between Augmented Reality using conventional perspective cameras versus non-central catadioptric cameras. We saw that, in theory, to be able to use augmented reality on non-central catadioptric cameras, one need to take into account changes on the following steps: projection of the 3D triangles to the 2D image plane; check for occlusions on the projected triangles; and compute the illumination associated to each triangle. After identifying and understanding these problems, we proposed changes to each of these steps. From the experimental results, we conclude that the proposed solutions work well with an acceptable computation effort.

Now, since we fully understand the differences between Augmented Reality using conventional perspective cameras and non-central catadioptric cameras, we can highlight some future work. The first is related to the projection of the triangles. We intentionally chose to use a large number of very small triangles to neglect the distortion effects associated with the projection of the 3D triangles. However, if this distortion can be considered, a smaller number of triangles could be used and the computation time would decrease significantly. Another improvement that we intend to consider are the shadows of the virtual object projected to the real scene, as well as the direct effect of the light source on the real scene.

References

- [Agrawal and Ramalingam, 2013] Agrawal, A. and Ramalingam, S. (2013). Single Image Calibration of Multi-Axial Imaging Systems. *IEEE Proc. Computer Vision and Pattern Recognition (CVPR)*.
- [Agrawal et al., 2011] Agrawal, A., Taguchi, Y., and Ramalingam, S. (2011). Beyond al-hazen'Problem: Analytical Projection Model for Non-Central Catadioptric Cameras with Quadric Mirrors. *IEEE Proc. Computer Vision and Pattern Recognition (CVPR)*.
- [Appel, 1968] Appel, A. (1968). Some techniques for shading machine renderings of solids. *Proceeding of the AFIPS*.
- [Azuma, 1997] Azuma, R. T. (1997). A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments: MIT Press Journal*.
- [Baker and Nayar, 1999] Baker, S. and Nayar, S. K. (1999). A Theory of Single-Viewpoint Catadioptric Image Formation. *International Journal of Computer Vision*.
- [Blinn, 1977] Blinn, J. F. (1977). Models of Light Reflection for Computer Synthesized Pictures. *ACM SIGGRAPH*.
- [Carpenter, 1984] Carpenter, L. (1984). The A-buffer, an antialiased hidden surface method. *Computer Graphics, Vol. 18, No. 3*.
- [Chen et al., 2009] Chen, I. Y.-H., MacDonald, B., and Wunsche, B. (2009). Mixed Reality Simulation for Mobile Robots. *IEEE Proc. Int'l Conf. Robotics and Automation (ICRA)*.
- [Chintamani et al., 2010] Chintamani, K., Cao, A., Ellis, R. D., and Pandya, A. K. (2010). Improved Telem manipulator Navigation During Display-Control Misalignments Using Augmented Reality Cues. *IEEE Trans. Systems, Man and Cybernetics, Part A: Systems and Humans*.
- [Debevec, 2008] Debevec, P. (2008). Rendering Synthetic Objects into Real Scenes: Bridging Traditional and Image-based Graphics with Global Illumination and High Dynamic Range Photography. *ACM SIGGRAPH 2008*.
- [Fournier et al., 1993] Fournier, A., Gunawan, A. S., and Romanzin, C. (1993). Common Illumination between Real and Computer Generated Scenes. *Proceeding of Graphics Interface (GI'93)*.
- [Fuchs et al., 2009] Fuchs, H., Livingston, M. A., Raskar, R., Colucci, D., Keller, K., State, A., Crawford, J. R., Rademacher, P., Drake, S. H., and Meyer, A. A. (2009). Augmented Reality Visualization for Laparoscopic Surgery. *International Conf. on Medical Image Computing and Computer Assisted Intervention (MICCAI)*.
- [Gonçalves, 2010] Gonçalves, N. (2010). On the reflection point where light reflects to a known destination in quadric surfaces. *Optics Letters*.
- [Goodrich and Schultz, 2007] Goodrich, M. A. and Schultz, A. C. (2007). Human-Robot Interaction: A Survey. *Foundations and Trends in Human-Computer Interaction*.
- [Gouraud, 1971] Gouraud, H. (1971). Continuous Shading of Curved Surfaces. *IEEE Trans. Comput*.
- [Grossberg and Nayar, 2003] Grossberg, M. D. and Nayar, S. K. (2003). A General Imaging Model and a Method for Finding its Parameters. *IEEE Proc. Int'l Conf. Computer Vision (ICCV)*.
- [Hartley and Zisserman, 2000] Hartley, R. and Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge University Press.
- [Hughes et al., 2014] Hughes, J., Dam, A. V., Mcguire, M., Skylar, D. F., Foley, J. D., Feiner, S. K., and Akeley, K. (2014). *Computer Graphics: Principles and Practice Third Edition*. Addison-Wesley, United States of America.
- [Micusik and Pajdla, 2004] Micusik, B. and Pajdla, T. (2004). Autocalibration & 3D Reconstruction with Non-central Catadioptric Cameras. *IEEE Proc. Computer Vision and Pattern Recognition (CVPR)*.
- [Nalwa, 1996] Nalwa, V. S. (1996). A True Omnidirectional Viewer. *Technical report, Bell Laboratories*.
- [Nayar and Baker, 1997] Nayar, S. K. and Baker, S. (1997). Catadioptric Image Formation. *Proceedings of the 1997 DARPA Image Understanding Workshop*.
- [Perdigoto and Araujo, 2013] Perdigoto, L. and Araujo, H. (2013). Calibration of mirror position and extrinsic parameters in axial non-central catadioptric systems. *Computer Vision and Image Understanding*.

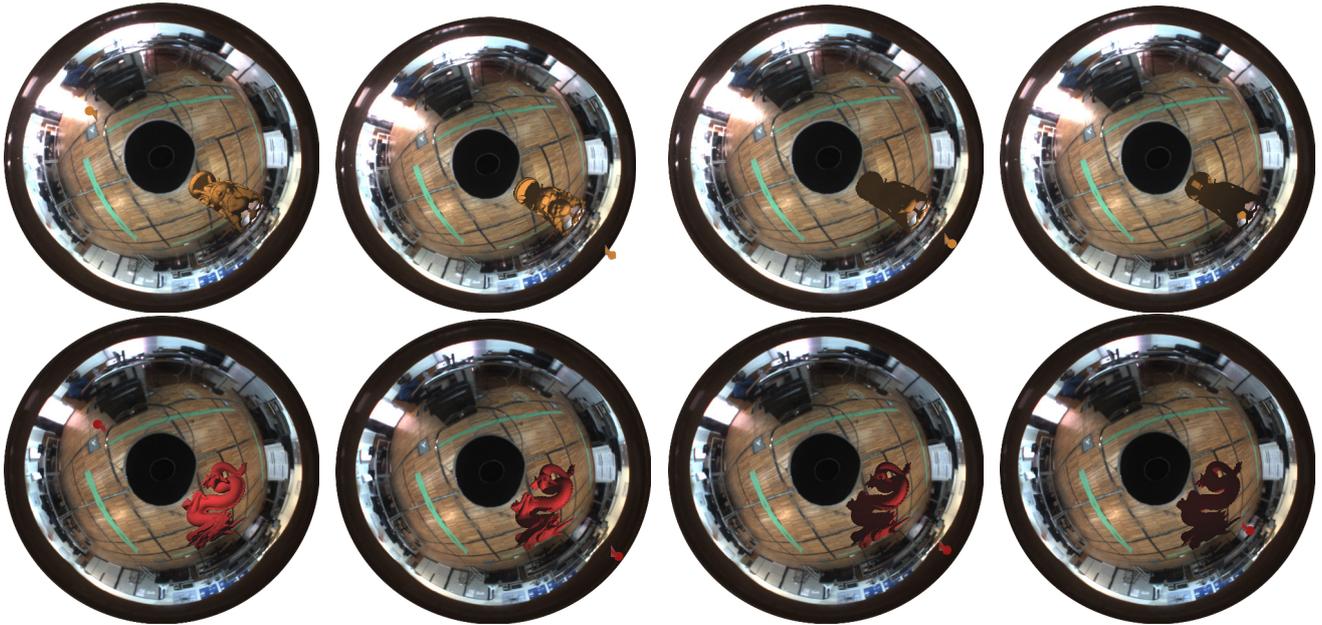


Figure 5: In this figure we show a set of frames, in which we apply the proposed framework, considering a moving spotlight. For this experiment, we used the following objects: the “happy buddha” (first row) and the “dragon” (second row). A video with the complete sequence is sent in supplementary material.

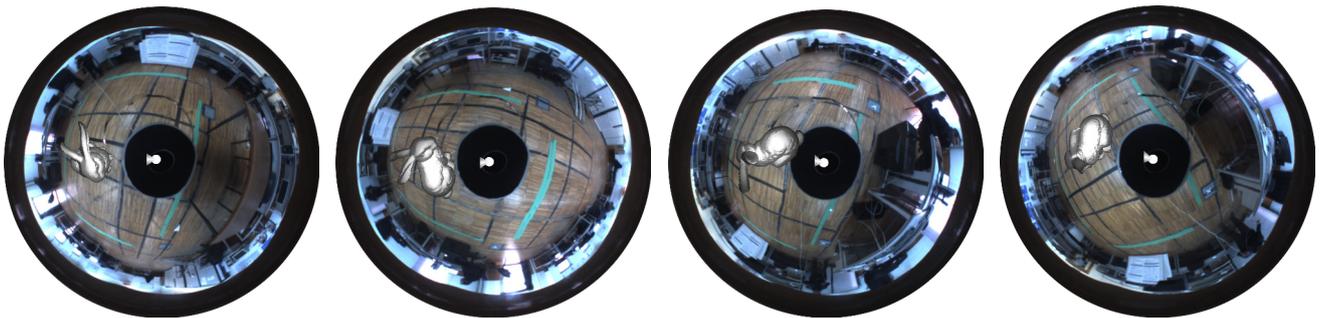


Figure 6: In this figure we show a set of frames, in which we apply the proposed framework, using a mobile robot. For this experiment, we used the Stanford “bunny”. A video (recorded in real-time) with the complete sequence is sent in supplementary material.

[Phong, 1975] Phong, B. T. (1975). Illumination for Computer Generated Pictures. *Commun. ACM*.

[Santos et al., 2012] Santos, A. L., Lemos, D., Lindoso, J. E. F., and Teichrieb, V. (2012). Real Time Ray Tracing for Augmented Reality. *IEEE Symposium on Virtual and Augmented Reality (SVR)*.

[Sato et al., 1999] Sato, I., Sato, Y., and Ikeuchi, K. (1999). Acquiring a Radiance Distribution to Superimpose Virtual Objects onto a Real Scene. *IEEE Transactions on Visualization and Computer Graphics*.

[Schweighofer and Pinz, 2008] Schweighofer, G. and Pinz, A. (2008). Globally Optimal $O(n)$ Solution to the PnP Problem for General Camera Models. *Proc. British Machine Vision Conference (BMVC)*.

[Stanford University CG Lab, 1993] Stanford University CG Lab (1993). Stanford Bunny. <https://graphics.stanford.edu/data/3Dscanrep/>.

[Swaminathan et al., 2001] Swaminathan, R., Grossberg, M. D., and Nayar, S. K. (2001). Caustics of Catadioptric Cameras. *IEEE Proc. International Conference on Computer Vision (ICCV)*.

[Swaminathan et al., 2003] Swaminathan, R., Grossberg, M. D., and Nayar, S. K. (2003). A Perspective on Distortions. *IEEE Proc. Computer Vision and Pattern Recognition (CVPR)*.

Especificação de Interrogações para Recuperação 3D em Ambiente Imersivo

Pedro B. Pascoal^{1,3}Daniel Mendes¹Alfredo Ferreira¹Miguel Sales Dias^{2,3}¹INESC-ID / IST / Universidade de Lisboa²ISCTE – Instituto Universitário de Lisboa / ISTAR-IUL, Lisboa, Portugal³Microsoft Language and Development Center, Lisboa, Portugal

pmbp@tecnico.ulisboa.pt, danielmendes@tecnico.ulisboa.pt,

alfredo.ferreira@inesc-id.pt, miguel.dias@microsoft.com

Resumo

O desenvolvimento de diversas ferramentas de modelação e captura de objetos 3D tem aumentado o volume de objetos 3D digitais acessíveis na internet. Consequentemente, a área de recuperação de modelos 3D tem sido cada vez mais explorada, como resposta aos desafios decorrentes de pesquisa destes objetos. Contudo, sistemas de pesquisa tradicionais não tiram partido de ambientes imersivos para pesquisar objetos 3D. Neste trabalho apresentamos um sistema de especificação de interrogações em ambiente imersivo, que permitisse uma interação mais natural. A interação com o sistema decorre em ambiente imersivo de realidade virtual, onde definem uma interrogação através de esboços 3D, para obter objetos semelhantes, recorrendo a mecanismos de pesquisa.

Palavras-Chave

Ambientes virtuais, esboços, especificação de interrogações para pesquisa, interação multimodal, recuperação de objetos 3D, visualização imersiva.

1. INTRODUÇÃO

Nos últimos anos, a quantidade de objetos 3D em repositórios digitais tem vindo a aumentar. Tal deve-se aos mais recentes avanços tecnológicos, que tem vindo a permitir criar objetos tridimensionais com maior facilidade. Esse rápido crescimento do volume dos repositórios 3D tem agravado a necessidade de métodos eficientes para pesquisa de objetos 3D.

Contudo, as soluções de pesquisa atuais ainda enfrentam grandes desafios e desvantagens. Entre outros, amplamente identificados [Datta08], destacamos dois. Em primeiro lugar, as pesquisas dependem de meta-informação, baseada em palavras-chave, o que faz com que sejam reduzidas a simples pesquisas por texto. De forma geral, estas pesquisas fazem uso de anotações, tais como o nome dos ficheiros, legendas, ou mesmo referências associadas, que podem não estar diretamente relacionadas com conteúdo media. Este conceito também se aplica na recuperação de objetos 3D [Funkhouser03]. Funkhouser et al., faz uso de sinónimos para aumentar as palavras possíveis para descrever objetos 3D e minimizar a escassez do vocabulário, contudo, as descrições disponíveis continuam a ser insuficientes, em especial relativamente à geometria dos objetos.

Como solução para esta limitação, podemos recorrer à pesquisa-por-exemplo [Zloof77], onde é usado objeto como interrogação, para encontrar objetos semelhantes em termos do aspeto visual, tal como a cor ou forma [Paquet97]. Contudo, este método de interrogação exige

que o utilizador tenha “*à priori*” um objeto semelhante ao procurado, o que nem sempre é o caso.

Em segundo lugar, os resultados são apresentados como uma lista de elementos. Esses elementos são geralmente imagens ilustrativas do conteúdo, ou apenas nomes. Tal metodologia dificulta enormemente a interpretação dos resultados da consulta, especialmente para dados media mais complexos, como são os objetos 3D. Por exemplo, numa avaliação de um conjunto de técnicas de seleção de melhor vista [Dutagaci10], é demonstrado que nenhum dos métodos analisados é consistente, sendo que todos fornece algumas imagens que não permitem a correta identificação do modelo 3D que representam. Em conclusão, imagens nem sempre fornecem toda a informação necessária para identificar o objeto 3D que está sendo visualizado.

Em conclusão, a maioria das soluções atuais falham ao não considerar a perceção humana e interação com o sistema. Por exemplo, no Princeton 3D Model Search Engine [Funkhouser03], os resultados são mostrados como uma lista de vistas 2D, o que dificulta muito a interpretação dos modelos. Adicionalmente, e embora ofereça um vasto conjunto de interfaces para a especificação de interrogação: pesquisa-por-texto; pesquisa-por-exemplo; pesquisa-por-esboço 2D e 3D, o uso de algumas interfaces, particularmente a de esboços 2D / 3D, é não trivial, sendo que a restrição a rato e janela, torna o seu uso limitado e carente de uma interação mais natural com o utilizador.

Neste trabalho, propomos uma nova abordagem para criação e pesquisa de objetos 3D, onde os objetos são exibidos em um ambiente de realidade virtual (VR), que permite uma completa percepção da sua forma.

Com o nosso trabalho, é utilizado o desenho no espaço 3D, como forma de criar e refinar objetos, usando um conjunto de técnicas multimodais. Com a nossa abordagem, pretendemos oferecer uma experiência de utilização que permita a utilizadores com pouca experiência conseguirem pesquisar objetos 3D com facilidade suficiente. Para isso, recorreremos a visualização imersiva, e interação natural baseada em gestos.

2. TRABALHO RELACIONADO

Ao contrário da pesquisa-por-exemplo [Zloof77], onde é necessário possuir um objeto que represente o objeto que se quer encontrar, a recuperação por esboço permite ao utilizador descrever o objeto e usar essa descrição como pesquisa para o obter objetos semelhantes.

Contudo, e embora a recuperação baseada em esboço seja um processo simples no contexto de recuperação por imagens, isso não é o caso em recuperação de objetos 3D. Para comparar um esboço 2D com objetos 3D é necessário mapear o objeto 3D para o espaço 2D. Uma das abordagens consiste em extrair um conjunto de vistas 2D, extraídas de diferentes pontos de vista. Um exemplo de tal abordagem é o trabalho de Saavedra et al. [Saavedra11], onde para cada objeto 3D, extraem 14 vistas de diferentes perspetivas. De seguida, para cada vista, calculam os contornos, usando a técnica “suggestive contours” [DeCarlo03], que cria um imagem não-fotorrealista, que se assemelha desenhos da mão do objeto 3D, e que será usada para a comparação com o esboço da interrogação.

Mais tarde, Shao et al. apresenta um outro método para recuperação de objetos 3D baseado em esboço [Shao11]. O esboço de interrogação é transformado em contornos vetoriais para permitir o cálculo de uma medida de similaridade local, sendo só consideram os traços próximos a detalhes importantes, ignorando distorções e traços insignificantes. Isto permite alguma tolerância a erros. Contudo, este tipo de soluções têm duas limitações.

Por um lado, dependem da eficiência do algoritmo que extrai as projeções 2D dos modelos 3D. Por outro, estes trabalhos assumem que os esboços se assemelham aos objetos 3D. Contudo, apesar, de em geral, todo o utilizador ser capaz de desenhar, a maioria não são artistas fiéis [Eitz12]. Estas soluções falham na oferta de uma interação mais natural para o utilizador.

Uma abordagem completamente diferente é proposta por Holz and Wilson [Holz11], que usam gestos para descrever e recuperar objetos de uma base de dados. Este trabalho consiste na captura dos gestos dos utilizadores, usando a forma e a movimentação das mãos para criar esboços tridimensionais. Os autores concluíram que os participantes conseguiam manter as proporções dos objetos físicos relativamente corretas e que, em zonas com mais detalhes, realizavam os gestos de forma mais lenta.

No entanto, a apresentação dos resultados neste trabalho também se limita ao uso da abordagem tradicional de listas de vistas.

Nesse contexto, Pascoal et al. apresenta um método de visualização de resultados de pesquisas onde os resultados são dispostos num ambiente virtual [Pascoal12]. Os utilizadores podem explorar e manipular os resultados de forma imersiva e mais natural, o que oferece uma melhor visualização e interpretação dos resultados, quando comparado com outras soluções mais tradicionais de recuperação 3D.

Mais recentemente, Henriques et al., realiza um estudo de diferentes técnicas para a consulta e visualização de resultados de pesquisas em ambientes virtuais, para entender quais métodos são mais natural [Henriques14]. Na especificação de uma pesquisa, o utilizador descreve o objeto pretendido, através do uso combinado de gestos e fala natural, sendo de seguida, apresentados um conjunto de resultados, onde é possível selecionar e manipular cada modelo 3D. No seu estudo são testados quatro modos de visualização, criados ao adaptar a abordagem grelha 2D tradicional a um ambiente imersivo 3D. Nesta avaliação concluí-se que o uso de ambientes 3D imersivos oferece melhores resultados no contexto dos objetos 3D, sendo menos suscetível a erros.

Finalmente, e apesar dos trabalhos apresentados, não existe até à data, nenhuma solução que unifique e possua todas as características consideradas importantes para uma melhor recuperação e visualização de objetos 3D. Com base na necessidade de tal solução que apresentamos o presente trabalho, permitindo explorar um campo ainda não explorado e contribuindo para o desenvolvimento de uma ferramenta mais poderosa, que combine as vantagens de cada um destes trabalhos.

3. SOLUÇÃO

O principal objetivo da nossa abordagem é permitir uma mais fácil especificação da interrogação em pesquisas de objetos 3D, através do uso de esboços no espaço. Após o desenho do esboço, a pesquisa de objetos 3D será feita através do uso de interrogações por exemplo, usando o objeto 3D que foi criado. No final, os objetos retornados serão visualizados num ambiente imersivo de realidade virtual tridimensional, Figura 1.

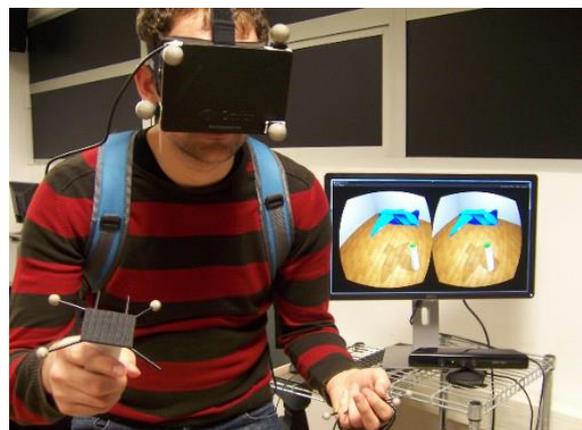


Figura 1. Utilizador a interagir com o sistema.

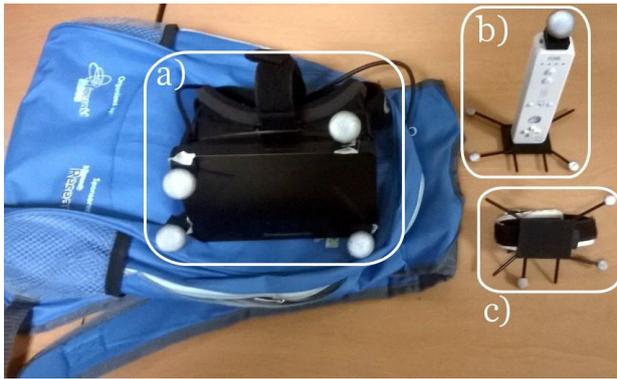


Figura 2. Equipamento utilizado na nossa abordagem. a) Oculus Rift; b) Comando Wiimote; c) Pulseira.

3.1 Configuração do Hardware

Para a nossa abordagem usamos um sistema de seguimento por marcadores, Optitrack, composto por dez câmaras Motion Capture da NaturalPoint. Em cada dispositivo que o utilizador usa estão colocados marcadores, que permitem calcular não só a sua posição no espaço tridimensional como também a sua orientação. são também utilizados óculos HMD de realidade virtual, Oculus Rift, para permitir a visualização em ambiente imersivo de realidade virtual. Usando marcadores colocados nestes óculos, foi possível detetar a posição do utilizador no mundo virtual. Adicionalmente, para permitir liberdade ao utilizador, usamos uma bateria e restantes equipamentos numa mochila de alças largas e almofadadas, permitindo conforto e liberdade de movimentos. Para criação de esboços, é utilizado um comando Wiimote. Os botões do comando permitem ainda interagir com o menu do sistema. Para representar o menu, é usada uma simples pulseira para a mão, com marcadores. Esta destina-se essencialmente para permitir localizar a outra mão do utilizador. Os equipamentos utilizados estão ilustrado na Figura 2.

3.2 Arquitetura

A arquitetura proposta para a solução, tem por base as arquiteturas dos trabalhos relacionados. Esta arquitetura, ilustrada na Figura 3, é composta por quatro módulos: Módulo de Tracking, Módulo de Visualização, Módulo de Interação e Módulo de Recuperação.

3.2.1 Módulo de Tracking e Visualização

O Módulo de Tracking capta os movimentos realizados pelo utilizador no espaço, enquanto realiza os esboços. Neste módulo é feita a captura de movimentos usando o Optitrack que, com o auxílio de marcadores, permite que sejam seguidos os movimentos feitos pela caneta enquanto o utilizador faz os seus esboços no espaço, retornando um conjunto de pontos. Este conjunto de pontos chega ao Módulo de Recuperação onde vai ser interpretado como um esboço 3D, que vai ser decomposto num conjunto de vistas 2D, cada uma paralela a um plano. Por sua vez, o Módulo de Recuperação irá devolver resultados para as interrogações que forem realizadas.

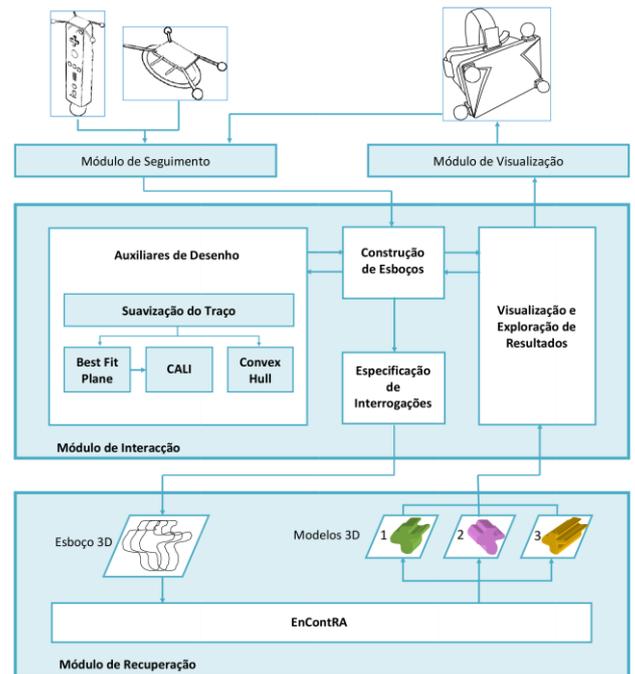


Figura 3. Arquitetura da solução proposta.

Os objetos resultado obtidos são depois apresentados pelo Módulo de Visualização, que é responsável por renderizar utilizando o motor gráfico Unity 3D, sendo visualizados no espaço tridimensional usando os Oculus Rift. Isto permite ao utilizador observar o ambiente em que realiza os esboços, bem como os resultados apresentados numa interrogação, dentro de um ambiente imersivo 3D.

3.2.2 Módulo de Interação

O Módulo de Interação será composto por quatro sub-módulos que permitem a comunicação com os outros módulos: especificar as interrogações, manipular e navegar pelos objetos e explorar os resultados de uma interrogação.

O sub-módulo “Construção de Esboços”, existente dentro do Módulo de Interação, é o que permite construir um objeto através de um esboço 3D. Durante a criação de um esboço, são utilizados diversos componentes, entre eles traços, que são a base para a criação do desenho. A partir de um traço os vários auxiliares de desenho calculam possíveis correções e sugestões para substituir o traço original, assim como possíveis polígonos e objetos sólidos, usando a informação espacial de todos os traços desenhados. É possível também utilizar sólidos pré-criados, adicionando-os ao esboço.

Este módulo recebe as posições das mãos do utilizador, e desta forma, processando os seus movimentos, rotações e localização espacial, encontra os pontos nos quais o esboço do utilizador deve estar, renderizando-o no ambiente de realidade virtual. Isto permite o desenho imediato dos traços desenhados, fazendo mapeamento direto da mão real do utilizador com a sua representação no mundo virtual, sem discrepâncias.

Quando o utilizador estiver satisfeito com o objeto que desenhado, é dada a ordem de pesquisa, e o modelo esboçado é enviado ao módulo “Especificação de Interrogações” que retira informações do objeto criado, para pesquisar por objetos semelhantes.

3.2.3 Módulo de Recuperação

O Módulo de Recuperação é responsável por indexar e devolver objetos 3D. Os objetos 3D utilizados os foram os do repositório do do Princeton Shape Benchmark [Shilane04]. Essa coleção tem um total de 1814 modelos, que foram recolhidos de diferentes fontes na internet.

Para a realização da recuperação de objetos 3D, foi utilizada a biblioteca ThOR [Pascoal15]. Esta biblioteca oferece um conjunto de ferramentas para a criação de soluções específicas para recuperação de objetos 3D.

Dessa biblioteca usamos o descritor “*Spherical Harmonic Descriptor*” [Kazhdan03] para extrair características de cada objeto. Aquando da interrogação, estas características indexadas são então comparadas com as do objeto da interrogação, permitindo devolver os resultados mais semelhantes.

3.3 Menu em Paleta

Na nossa abordagem o Menu é representado usando uma metáfora da paleta de um pintor. Assim como um pintor dispõe as cores que quer utilizar numa tela, nós dispomos as funcionalidades para criar esboços, Figura 4.

Interagindo com os diferentes ícones do menu é possível escolher várias definições para os traços que podem ser desenhados pelo utilizador. Há a possibilidade de alterar a cor do traço e ainda a sua espessura.

É ainda possível selecionar sólidos pré-criados para colocar na cena. O utilizador pode escolher entre um cilindro, uma esfera, um paralelepípedo e uma pirâmide. Estes destinam-se a auxiliar a criação de um esboço mais completo, permitindo um menor esforço por parte do utilizador.

Cada uma destas funcionalidades do menu está identificada pelo seu ícone (Figura 4). Este tipo de menu proporciona uma experiência familiar e natural aos utilizadores, sendo fácil aceder às várias funcionalidades do sistema.

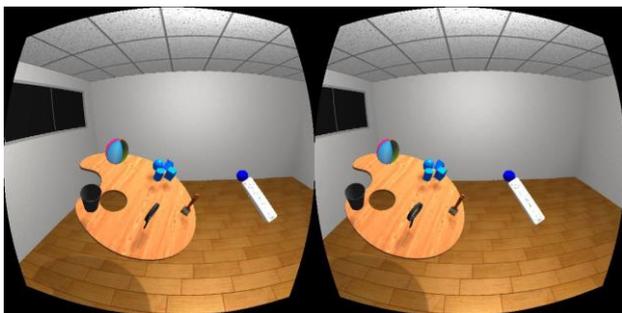


Figura 4. Menu no ambiente virtual.

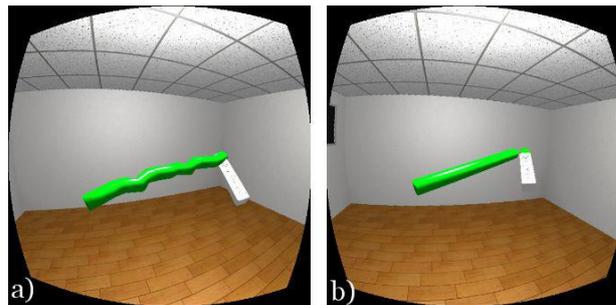


Figura 5. a) sem suavização e b) com suavização.

3.4 Auxiliares de Desenho

Como já referido na secção 2, apesar de todo o utilizador ser capaz de desenhar, a maioria não são artistas fiéis [Eitz12]. Adicionalmente, sendo esboços feitos no espaço, têm complexidade acrescida, sendo muitas vezes possível o utilizador não conseguir criar traços estritamente direitos, dadas oscilações e movimentos ligeiros da mão. Como tal, é essencial oferecer um conjunto de ferramentas que assistam o utilizador na criação do esboço de forma fácil e interativa. Tais tarefas são realizadas pelo módulo Auxiliares de Desenho, existente dentro do Módulo de Interação. Após o desenho de cada traço, é possibilitado ao utilizador corrigir o traço do seu esboço. Ao concluir cada traço, é aplicada uma suavização do traço (Figura 5), e são apresentadas cinco sugestões ao utilizador.

3.4.1 Suavização do traço

A primeira sugestão é a simples possibilidade de manter o que está esboçado até ao momento, sem nenhuma alteração. A Suavização do Traço permite criar linhas no esboço completamente direitas, sem nenhuma perturbação. Quando o utilizador começa a esboçar, os dois primeiros pontos que cria servem de base para o traço que vai criar. Estes dois primeiros pontos criam o segmento inicial. A partir do terceiro ponto, cada ponto seguidamente desenhado é verificado de acordo com o vetor direção da reta, formado pelos dois pontos iniciais. Caso a linha seja a mesma, e mantendo sempre o ponto inicial A da linha, é encontrado um novo ponto P que será o último da linha, de acordo com a seguinte equação:

$$P = A + \hat{AB} \times \sqrt{(C_x - A_x)^2 + (C_y - A_y)^2 + (C_z - A_z)^2}$$

Com este novo último ponto encontrado, o sistema pega na linha já criada e altera o seu último ponto, substituindo pelo novo. Na Figura 5 é possível ver dois traços esboçados, um sem suavização e o outro com suavização.

3.4.2 Best Fit Plane

Para segunda e terceira sugestões é usado o algoritmo Best Fit Plane. São usados os traços esboçados pelo utilizador para encontrar o melhor plano aproximado que contém os traços, projetando-os nele e recortando um polígono no plano para a segunda sugestão (Figura 6), e projetando o traço no plano para a terceira sugestão, criando um novo traço (Figura 7).

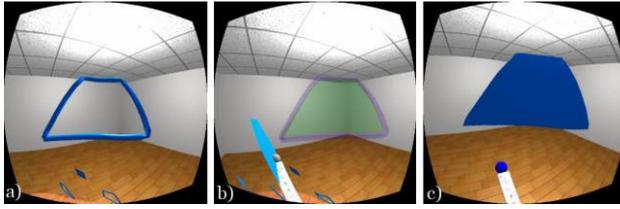


Figura 6. Polígono criado a partir de um esboço, através do Best Fit Plane: a) esboço, b) pré-visualização da substituição; c) esboço substituído por polígono resultante.

O algoritmo é composto pelos seguintes passos. Após ser criado um objeto vazio o sistema encontra todos os pontos presentes no esboço são usado para criar objeto 3D. Estes são dados pela ordem dos ponteiros de um relógio. Este passo é importante uma vez que determina qual a direção na qual a normal do plano irá apontar.

Assim, dados m pontos (x_i, y_i, z_i) , a normal (n_x, n_y, n_z) é encontrada pela seguinte equação:

$$n_x = \sum_{i=1}^m (z_i + z_{i+1})(y_i - y_{i+1})$$

$$n_y = \sum_{i=1}^m (x_i + x_{i+1})(z_i - z_{i+1})$$

$$n_z = \sum_{i=1}^m (y_i + y_{i+1})(x_i - x_{i+1})$$

Este vetor é então normalizado. De seguida, é encontrado o valor da distância média d de cada ponto ao plano. Para tal faz-se o somatório de todos os pontos, encontrando um ponto médio, que depois faz o produto interno com a normal n :

$$d = \frac{1}{m} \left(\sum_{i=1}^m p_i \right) \cdot n$$

O próximo passo, tendo obtido a normal e a distância, é criar um plano matemático. Este não tem qualquer representação visual. Para poder criar um polígono, é preciso realizar a translação de todos os pontos para este plano matemático. É encontrada a distância escalar s de cada ponto ao plano matemático e a partir daí pode-se encontrar a projeção q de cada ponto p :

$$q = p - s \times n$$

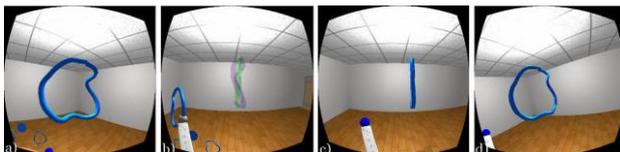


Figura 7. Traço criado a partir de um esboço, através do Best Fit Plane: a) esboço; b) pré-visualização lateral da substituição; c) esboço substituído por traço resultante; d) traço visto frontalmente.

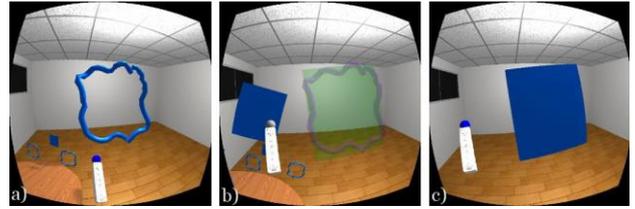


Figura 8. Resultado do CALI criado a partir de um esboço: a) esboço; b) pré-visualização da substituição; c) esboço substituído por polígono resultante.

Após obter o conjunto dos pontos mapeados para o plano, verifica-se se não existem pontos repetidos. Por fim, a partir deste conjunto de pontos, cria-se uma malha que corresponde a um polígono planar que contém a projeção de todos os pontos do esboço criado pelo utilizador. Cria-se ainda um traço que corresponde ao tubo original rebatido no plano encontrado.

Estando concluído o Best Fit Plane, são criados os modelos com os resultados do algoritmo, que são depois importados pelo sistema para a cena.

3.4.3 CALI

Após usar o Best Fit Plane para encontrar o melhor plano aproximado que contém os traços esboçados, utilizando uma modificação da ferramenta CALI [Fonseca02] no plano encontrado. O resultado é um polígono com a forma geométrica devolvida pela ferramenta CALI.

O algoritmo CALI é uma ferramenta que permite reconhecer esboços bidimensionais, e identificar diversas formas, tais como triângulos, retângulos e diamantes.

O resultado é um polígono com a forma geométrica devolvida pela ferramenta CALI (Figura 8).

3.4.4 Convex Hull

Por fim, para a quinta sugestão é realizado o algoritmo Convex Hull no esboço, tendo como resultado um sólido [Barber96].

Para tal, o sistema usa o conjunto dos pontos existentes no esboço e cria o menor conjunto convexo que contém todos os pontos. Após ter todos os pontos, o algoritmo encontra os pontos com coordenadas mínimas e máximas para cada uma das 3 dimensões: x ; y ; e z . Estes pontos fazem parte do Convex Hull resultante.

De seguida, é criado um *simplex* inicial, a partir de 3 + 1 pontos, ou seja, dado o espaço do sistema ser tridimensional, é criado um tetraedro regular, com três pontos na base e um no topo do sólido, como uma pirâmide. Do grupo de pontos encontrados anteriormente, com maiores e menores coordenadas, são usados os dois pontos mais distantes entre si, sendo o terceiro o ponto mais afastado da linha definida pelos dois primeiros pontos. Este é o triângulo da base. Para o topo da pirâmide, é encontrado, a partir da lista de todos os pontos, o ponto mais distante do triângulo da base. A partir deste *simplex* vai ser construído o Convex Hull final.

São verificados os restantes pontos. Para cada ponto verifica-se se o mesmo se encontra dentro do *simplex* encontrado (por trás de todas as faces do *simplex*). Se sim, é ignorado e não é utilizado no processo. Caso contrário, o ponto é atribuído a uma face. O ponto é atribuído à primeira face que “consegue ver”. Assim, cada ponto é dado a apenas uma face. Cada face contém o seu conjunto de pontos.

Seguidamente, para cada face, do seu conjunto de pontos atribuídos é encontrado o ponto que se encontra mais distante. Verificam-se todas as faces adjacentes à atual que são “vistas” a partir do ponto.

Isto permite determinar as arestas no horizonte, ou seja, as arestas que delimitam a parte visível do Convex Hull atual da parte não visível, do ponto de vista do ponto atual.

Cada aresta horizonte cria com o ponto atual um novo triângulo. Novas faces são criadas e adicionadas ao Convex Hull atual. Os pontos que não foram utilizados de cada uma das faces existentes anteriores são reatribuídos às faces novas, novamente à primeira face que “conseguem ver”. São descartados se se encontrarem atrás de todas as faces.

O algoritmo repete os passos realizados nos dois anteriores parágrafos até esgotar os pontos do esboço, ficando encontrado o Convex Hull. Finalmente, é criado um modelo com o resultado do algoritmo, que é depois importado pelo sistema para a cena. Na Figura 9 é possível ver o Convex Hull formado a partir de um esboço do utilizador.

3.5 Modelação e Integração de Técnicas

Como detalhado na secção anterior, a nossa abordagem permite desenhar traços, formas, objetos e até cenários, usando esboços. Cada esboço do utilizador pode ser transformado em um objeto tridimensional e, em conjunto, fazer parte de um cenário. Para a criação de esboço, o utilizador desenha traços, que com a ajuda dos auxiliares de desenho consegue aperfeiçoar para traços retos, polígonos e sólidos. Cada uma das técnicas cria elementos que permitem ao utilizador modelar o seu esboço, e apesar de poderem ser usadas isoladamente, sendo que quando combinados permite criar objetos 3D completos.

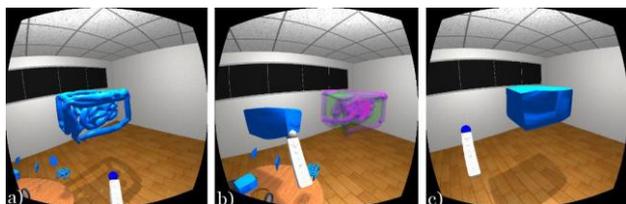


Figura 9. Convex Hull criado a partir de um esboço: a) esboço; b) pré-visualização da substituição; c) esboço substituído por sólido resultante.

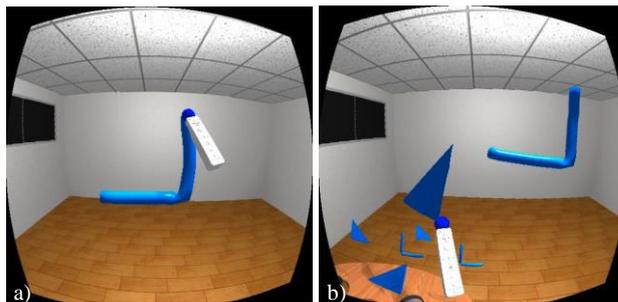


Figura 10. a) Utilizador começa um esboço. b) O utilizador vê a sugestão de um polígono do Best Fit Plane.

A base de todos os algoritmos passa pela Suavização do Traço. Estando ativo, recebe o traço realizado pelo utilizador, que pode ter algumas distorções por ter sido criado no espaço tridimensional.

Como também já mencionado, o algoritmo CALI depende dos cálculos executados pelo Best Fit Plane para encontrar um plano. Tal torna a sua execução mais rápida ao reutilizar planos e pontos encontrados previamente. Especificamente, são reutilizados o plano e os pontos transacionados para o mesmo, para descobrir qual a forma que mais se assemelha ao esboço.

Com a interligação destas ferramentas, o utilizador pode criar e completar o seu esboço, escolhendo a sugestão resultante de cada uma delas a partir da lista de sugestões. Cada sugestão obtida no componente Auxiliares de Desenho permite ir construindo o esboço progressivamente, sendo este visualizado interactivamente e inspecionado na Visualização e Exploração de Resultados (Figura 10).

Quando concluído o esboço, o utilizador pode usar o modelo 3D criado, como interrogação para uma pesquisa-por-exemplo à base de dados. De forma semelhante à visualização de sugestões, os resultados da pesquisa são apresentados na Paleta (Figura 11). Assim, como no caso das sugestões, antes de decidir escolhe substituir o esboço por um dos resultados, o utilizador pode pré-visualizar um resultado, sendo este apresentado sobreposto ao esboço, com as mesmas proporções do esboço. O utilizador vê desta forma como ficará o resultado se o escolher, e caso opte pode seleccionar e substituir esboço.

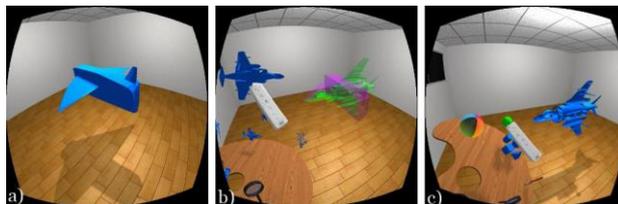


Figura 11. Esboço e pesquisa de um avião, utilizando a nossa abordagem: a) esboço criado; b) pré-visualização da substituição do esboço por um resultado; c) esboço substituído pelo resultado.

4. CONCLUSÕES E TRABALHO FUTURO

Nos últimos anos, o número de objetos 3D disponíveis digitalmente tem crescido. Contudo, soluções atuais para pesquisa de objetos 3D, ainda têm alguns desafios por resolver. Com o objetivo de ultrapassar as limitações presentes nos sistemas atuais, propôs-se um sistema imersivo para especificação de interrogações e recuperação 3D, usando esboços criados num espaço 3D.

Para o desenvolvimento da nossa solução utilizamos um conjunto de equipamentos que permitissem visualizar, especificar interrogações e interagir com os resultados num ambiente imersivo de realidade virtual. Foram utilizadas técnicas de interação multimodal que permitissem um método mais natural para a especificação de interrogações de objetos 3D para recuperação de objetos.

Com a nossa abordagem permitimos a utilizadores casuais, facilmente construírem um objeto através do desenho de um esboço 3D. Isto foi possível através da integração de vários componentes, que funcionam como auxiliares de desenho, e em conjunto oferecem ferramentas que assistem o utilizador na criação do esboço de forma fácil e interativa.

Apesar de se terem alcançado os objetivos propostos, acredita-se ainda ser possível melhorar e completar este trabalho. Adicionalmente, será necessária uma validação da abordagem desenvolvida, comparativamente com um sistema tradicional existente. Contudo, durante a realização de um conjunto de testes preliminares, a nossa abordagem permitiu não só criar representações mais completas dos objetos de forma mais rápida e simples. Como ponto negativo, foi notório o cansaço visual, que se mostrou significativamente elevado, um problema típico dos óculos HMD. Contudo, grau de diversão foi bastante positivo, e os auxiliares de desenho foram, em geral, considerados bastante úteis.

Com a abordagem demonstrada neste trabalho e os avanços alcançados, acreditamos que o nosso contributo possa servir de base para melhorar e impulsionar trabalhos futuros no mesmo tópico.

5. AGRADECIMENTOS

O trabalho apresentado neste artigo foi parcialmente financiado pela Fundação para a Ciência e a Tecnologia (FCT) através dos projectos TECTON-3D (PTDC/EEI-SII/3154/2012), UID/CEC/50021/2013 e da bolsa SFRH/BD/91372/2012.

6. REFERÊNCIAS

- [Barber96] C. B. Barber, D. P. Dobkin, & H. Huhdanpaa. (1996). The Quickhull Algorithm for Convex Hulls. *ACM Trans. Math. Softw.*, 22(4), 469–483. <http://doi.org/10.1145/235815.235821>
- [Datta08] R. Datta, D. Joshi, J. Li, & J. Z. Wang. (2008). Image retrieval: Ideas, influences, and trends of the

new age. *ACM Comput. Surv.*, 40(2), 5:1–5:60. <http://doi.org/http://doi.acm.org/10.1145/1348246.1348248>

- [DeCarlo03] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, & A. Santella. (2003). Suggestive Contours for Conveying Shape. In *ACM SIGGRAPH 2003 Papers* (pp. 848–855). New York, NY, USA: ACM. <http://doi.org/10.1145/1201775.882354>
- [Dutagaci10] H. Dutagaci, C. P. Cheung, & A. Godil. (2010). A benchmark for best view selection of 3D objects. In *Proceedings of the ACM workshop on 3D object retrieval* (pp. 45–50). New York, NY, USA: ACM. <http://doi.org/http://doi.acm.org/10.1145/1877808.1877819>
- [Eitz12] M. Eitz, J. Hays, & M. Alexa. (2012). How Do Humans Sketch Objects? *ACM Trans. Graph.*, 31(4), 44:1–44:10. <http://doi.org/10.1145/2185520.2185540>
- [Funkhouser03] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, & D. Jacobs. (2003). A search engine for 3D models. *ACM Trans. Graph.*, 22(1), 83–105. <http://doi.org/http://doi.acm.org/10.1145/588272.588279>
- [Henriques14] D. Henriques, D. Mendes, P. Pascoal, I. Trancoso, & A. Ferreira. (2014). Poster: Evaluation of immersive visualization techniques for 3D object retrieval. In *3D User Interfaces (3DUI), 2014 IEEE Symposium on* (pp. 145–146). <http://doi.org/10.1109/3DUI.2014.6798862>
- [Holz11] C. Holz, & A. Wilson. (2011). Data miming: inferring spatial object descriptions from human gesture. In *Proc. of the 2011 annual conference on Human factors in computing systems* (pp. 811–820). ACM.
- [Kazhdan03] M. Kazhdan, T. Funkhouser, & S. Rusinkiewicz. (2003). Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (pp. 156–164). Aire-la-Ville, Switzerland, Switzerland: Eurographics Association. Retrieved from <http://dl.acm.org/citation.cfm?id=882370.882392>
- [Paquet97] E. Paquet, & M. Rioux. (1997). Nefertiti: a query by content software for three-dimensional models databases management. In *Proceedings of the International Conference on Recent Advances in 3-D Digital Imaging and Modeling* (p. 345–). Washington, DC, USA: IEEE Computer Society. Retrieved from <http://dl.acm.org/citation.cfm?id=523428.825366>

- [Pascoal15] P. B. Pascoal, & A. Ferreira. (2015). ThOR: Three-dimensional Object Retrieval Library. In I. Pratikakis, M. Spagnuolo, T. Theoharis, L. Van Gool, & R. Veltkamp (Eds.), *Eurographics Workshop on 3D Object Retrieval*. The Eurographics Association. <http://doi.org/10.2312/3dor.20151058>
- [Pascoal12] P. B. Pascoal, A. Ferreira, & J. Jorge. (2012). Im-O-Ret: Immersive object retrieval. In *Virtual Reality Short Papers and Posters (VRW), 2012 IEEE* (pp. 121–122). <http://doi.org/10.1109/VR.2012.6180912>
- [Saavedra11] J. Saavedra, B. Bustos, M. Scherer, & T. Schreck. (2011). STELA: sketch-based 3D model retrieval using a structure-based local approach. In *Proc. ACM International Conference on Multimedia Retrieval (ICMR'11)* (pp. 26:1–26:8). ACM.
- [Shao11] T. Shao, W. Xu, K. Yin, J. Wang, K. Zhou, & B. Guo. (2011). Discriminative Sketch-based 3D Model Retrieval via Robust Shape Matching. *Comput. Graph. Forum*, 2011–2020.
- [Shilane04] P. Shilane, P. Min, M. Kazhdan, & T. Funkhouser. (2004). The Princeton Shape Benchmark. *Shape Modeling and Applications, International Conference on*, 0, 167–178. <http://doi.org/http://doi.ieeecomputersociety.org/10.1109/SMI.2004.1314504>
- [Zloof77] M. M. Zloof. (1977). Query-by-Example: A data base language. *IBM Systems Journal*, 16(4), 324–343. <http://doi.org/10.1147/sj.164.0324>

Malhas Segmentadas com Resolução Variável

Rui S. V. Rodrigues
ESTGV-IPV & U. da Beira Interior
Viseu & Covilhã
rsrodrigues@estv.ipv.pt

José F. M. Morgado
ESTGV-IPV
Viseu
fmorgado@estv.ipv.pt

Abel J. P. Gomes
Universidade da Beira Interior
Covilhã
agomes@di.ubi.pt

Resumo

As malhas com resolução variável (“multiresolution meshes”) têm sido utilizadas como alternativa à técnica dos LODs em ambientes virtuais e jogos de computador, em que o detalhe da cena depende da distância ao observador, bem como na (des)compressão de malhas em tarefas de transferência e transmissão mais célere de malhas através da Internet e da web, o que permite o carregamento progressivo da malha do lado do cliente, que começa com uma malha mais grosseira e termina com uma malha mais refinada. A principal contribuição deste artigo reside na utilização do conceito de multirresolução em malhas segmentadas, em vez de malhas simples (sem segmentação).

Palavras-Chave

Malhas triangulares, multirresolução, segmentação, simplificação de malhas, refinamento de malhas.

1. INTRODUÇÃO

As malhas de resolução variável (também, designadas por malhas multirresolução) surgiram como alternativa às malhas LOD (*level of detail*), por causa da poupança que se obtém de espaço em memória. Na verdade, enquanto um LOD é uma sequência finita de malhas com resoluções diferentes de um mesmo objeto residente em memória, uma malha de resolução variável requer só uma malha do mesmo objeto em memória. No entanto, uma malha multirresolução pode ser refinada e simplificada através de operadores de refinamento e de simplificação, respetivamente. Como dizia Garland [Garland 97], independentemente do nível de detalhe, uma malha de resolução variável deve manter ou, quando muito, aumentar ligeiramente as necessidades de espaço em memória relativamente ao espaço de memória ocupada pelo malha com mais detalhe.

Uma das preocupações sempre presentes para quem pretende desenhar e implementar um esquema de multirresolução prende-se com a preservação da forma do objeto que a malha representa, em particular quando se procede à simplificação da malha. De facto, o processo de simplificação tende a esbater as concavidades da malha. Para obviar este problema, utiliza-se normalmente um critério de manutenção de forma; por exemplo, a coalescência de dois triângulos que incidem numa aresta só tem lugar quando o ângulo diedral é zero ou próximo de zero. Neste artigo, a ideia não é usar um critério genérico de preservação da forma do objeto, mas sim utilizar a segmentação da malha em partes como mecanismo de preservação da forma. Noutras palavras, preserva-se a forma de uma malha preservando a sua segmentação em

partes.

Por isso, pode ser importante tentar perceber como os seres humanos percecionam os objetos e a sua forma. Como foi observado por Biederman [Biederman 87], o ser humano perceciona um dado objeto como um conjunto de partes. Já Hoffman [Hoffman 97] concluiu que o ser humano define as fronteiras entre as diferentes partes dos objetos pelas zonas de curvatura negativa mínima. Falar de um objeto como um agregado de várias partes significa *a priori* a existência de uma malha segmentada em diferentes regiões, tendo-se para isso utilizado um determinado critério de segmentação [Attene 06]. Veja-se o levantamento do estado da arte levado a cabo por Shamir em [Shamir 08] para mais informação sobre critérios e técnicas de segmentação de malhas trianguladas.

Com base na asserção acima referida, como sendo devida a Hoffman [Hoffman 97], Rodrigues et al. [Rodrigues 15] propuseram o algoritmo HPO (*human perception-oriented segmentation*), o qual segmenta a malha que representa um dado objeto em conformidade com a percepção em partes que o ser humano tem do dito objeto, pelo que se se quiser preservar a forma deve-se preservar as partes (ou segmentos); algumas malhas segmentadas através do algoritmo HPO podem ser vistas na Figura. 1.

A técnica de preservação da forma descrita neste artigo para malhas segmentadas com resolução variável tem por objetivo não só preservar os ciclos de arestas que separam as regiões (ou segmentos) da malha entre si, mas também os caminhos vincados (ou vincos), os ápices e os pontos máximos de curvatura (por exemplo as extremidades dos dedos de uma mão) existentes na malha, que são facilmente identificados pelos valores elevados da sua curva-

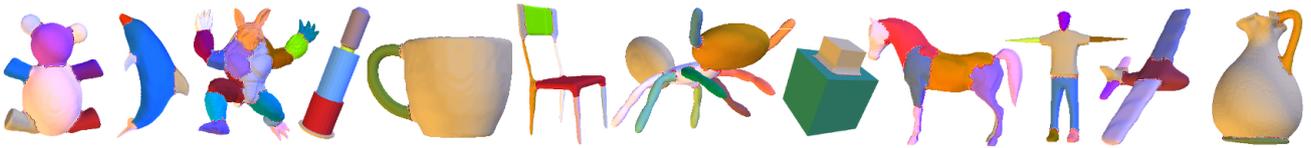


Figura 1. Malhas segmentadas com o algoritmo HPO.

tura, os quais —como refere Mangan et al. [Mangan 99]— são calculados através da norma da matriz de covariância.

Assim sendo, as principais contribuições deste artigo são as seguintes:

- Tanto quanto julgamos saber, propõe-se a primeira estrutura de dados para malhas *segmentadas* com resolução variável.
- Um algoritmo de simplificação que usa a informação da segmentação para manter a forma da malha. É preservada a fronteira entre as diferentes regiões da malha de forma a preservar os segmentos da malha.
- Preserva-se ainda os vincos e os pontos característicos (por exemplo, ápices) da malha.
- O refinamento e a simplificação podem ser aplicados a toda a malha ou somente a um conjunto específico de segmentos da malha.

O restante artigo está organizado da seguinte forma. Na secção 2 faz-se um resumo do trabalho relacionado existente. Na secção 3 é explicada a estrutura de dados proposta para suportar as malhas segmentadas de resolução variável. Nas secções 4 e 5 estão explicados (respetivamente) os algoritmos de simplificação e refinamento. Na secção 6 são apresentados os resultados do trabalho. E, finalmente, na secção 7 são feitas algumas considerações finais sobre o trabalho realizado.

2 TRABALHO RELACIONADO

Na literatura, existem duas famílias principais de malhas de resolução variável: subdivisão recursiva e subdivisão não-recursiva. Na subdivisão recursiva dividem-se os polígonos recursivamente todos da mesma maneira. Por exemplo, todo e qualquer triângulo é dividido em três ou quatro triângulos mais pequenos, de maneira que a malha vai ficando visualmente cada vez mais suave *quadtree* [Schröder 00] [Floriani 02]. Na subdivisão não-recursiva utiliza-se *a priori* uma malha com resolução máxima, a qual poderá então ser simplificada, bem como ser refinada se a resolução da malha não estiver no nível máximo [Garland 99]. Além disso, no caso da subdivisão não-recursiva, nem todos os triângulos (ou quadrângulos) serão sujeitos a refinamento/simplificação em cada nível de detalhe ou resolução.

No presente artigo, usar-se-á um esquema de multirresolução não recursiva. Normalmente, um esquema de multirresolução não recursiva utiliza a estrutura de dados para duas finalidades, nomeadamente:

- *Repositório das malhas de resolução variável.* O armazenamento de uma malha de resolução variável em memória serve dois propósitos, isto é, permite fazer operações de edição e operações de visualização.
- *Repositório das alterações das malhas.* Com este repositório guarda-se o historial das alterações, o que permite avançar ou voltar a uma malha específica com maior ou menor resolução.

Segundo Silva e Gomes [Silva 03], a natureza dos elementos que constituem uma malha tem um impacto significativo no *desempenho* das estruturas de dados das malhas de resolução variável. Uma forma de melhorar o desempenho é através de definição de esquemas de incidência entre os diferentes elementos da malha (vértices, arestas e faces). Ni e Bloor [Ni 94] chegaram à conclusão que a estrutura de dados simétrica C_4^9 é ótima para malhas bi-dimensionais. E este tipo de esquema de incidência já o encontramos nas estruturas de dados como a AIF [Silva 03] e GC (*ghost-cell* ou *aresta fantasma*) [Rodrigues 07].

Nos modelos de multirresolução não-recursiva, temos que codificar explicitamente as alterações e as dependências da malha na estrutura de dados. Essa codificação depende do tipo de elemento da malha que é modificado [Floriani 02], o que nos leva à seguinte classificação: multirresolução baseada em operações sobre arestas e multirresolução baseada em operações sobre vértices. No primeiro caso, utiliza-se operações de contração da aresta (*edge collapse*) e de divisão do vértice (*vertex split*) [Hoppe 96] [ES99] [Danovaro 05], ao passo que no segundo caso utiliza-se operações de inserção e remoção de vértices [Schroeder 92] [Ciampalini 96]) [Rossignac 93] [Luebke 97].

O modelo de multirresolução proposto utiliza um algoritmo de simplificação baseado na operação de contração da aresta. Por isso, a aresta a contrair deve ser selecionada segundo um determinado critério, nomeadamente: a chamada função de energia usada em *Mesh Optimization* [Hoppe 93] e otimizada nas *Progressive Meshes* [Hoppe 96]; minimização do erro associado a cada vértice usada em *Surface Simplification Using Quadric Error Metrics* [Garland 97]; já em *Normal-based Simplification Algorithm* (NSA) [Silva 04] escolhe-se a aresta a contrair com base na variação das normais às faces em redor da aresta alvo.

Em cenas 3D, é comum aplicar-se multirresolução às partes visíveis das malhas que representam objetos existentes numa dada cena, em particular no con-

texto de compressão e transmissão de geometria [Kim 01] [Yang 04] [Kim 06] [Maglo 11] [Maglo 13]. Aliás, é comum também usar-se a distância ao observador como critério de multirresolução [Hoppe 97] [ES99] [Pajarola 01] [Choe 09], em particular em modelos de terrenos. Neste sentido, poderá dizer-se que existem partes com mais ou menos detalhe, que poderiam ser designados por segmentos não fosse o facto de que as partes visíveis da malha dependerem da posição do observador. Quando muito, estas partes da malha que não são perceptualmente significativas poderão ser designadas por regiões (*charts*, do inglês).

Nos trabalhos anteriores não é tido em conta qualquer relação entre as regiões e as partes significativas dos objetos (i.e., segmentos). Este tipo de relação encontra-se plasmada pela primeira vez no trabalho de Cheng et. al. [Cheng 07], em que os segmentos são delineados pelos principais contornos de curvatura mínima da malha, embora este trabalho se tenha inspirado no trabalho de Yang et al. [Yang 04], que dividiam a malha em regiões perceptualmente não-significativas. Refira-se, no entanto, que Cheng e seus colegas não tiveram em conta a preservação das fronteiras dos segmentos.

Ao invés, o esquema de multirresolução aqui proposto serve o propósito de malhas segmentadas, em que se preserva a forma através da preservação das fronteiras dos segmentos. Além do mais, é possível ter diferentes níveis de multi-resolução ou de detalhe em diferentes segmentos, mesmo entre segmentos adjacentes ou vizinhos.

3 ESTRUTURA DE DADOS BASEADA NA CÉLULA FANTASMA

Neste trabalho, utilizou-se uma estrutura de dados baseada na célula fantasma (*ghost-cell*, do inglês) que foi desenvolvida por Rodrigues et al. [Rodrigues 07]. A estrutura de dados GC (*ghost-cell*) é uma estrutura de dados ótima do tipo C_4^9 , que codifica as seguintes relações topológicas: $V \prec E$ e $E \prec F$, e as suas relações inversas, $E \succ V$ e $F \succ E$.

Esta estrutura de dados foi estendida de molde a suportar malhas segmentadas, pelo que a esta variante da GC foi atribuída a designação de xGC (*extended GC*). As alterações introduzidas na xGC foram as seguintes:

- *Class Mesh*. Nesta classe foi incluída uma lista de segmentos.
- *Class Segment*. Esta é uma nova classe, a qual inclui o identificador do segmento e uma lista de referências para as faces que fazem parte do segmento.
- *Class Face*. Esta classe inclui uma referência para o segmento a que a face pertence.
- *Class Edge*. Nesta classe foi incluída uma *flag* para indicar se a aresta pertence à fronteira de um segmento ou não.
- *Class Vertex*. Nesta classe foi incluída uma *flag* para indicar se o vértice pertence ou não à fronteira do segmento.

Recorde-se que o carácter de resolução variável da estrutura de dados GC advém da existência de duas estruturas genealógicas, uma para os vértices e outra para as arestas. Também estas estruturas genealógicas sofreram as seguintes modificações:

- *Genealogia do vértice*. Nesta estrutura introduziu-se um novo campo para guardar informação sobre o tipo de contração de aresta que esteve na origem do vértice.
- *Genealogia da aresta*. Nesta estrutura foi otimizada a utilização da memória, porque a partir de certos níveis de simplificação os vértices fronteira da aresta sobrepõem-se (resultam da contração da aresta).

4 ALGORITMO SIMPLIFICAÇÃO

No algoritmo implementado aplica-se a operação de contração da aresta para conseguir estes objetivos. Para a simplificação é necessário percorrer o array de arestas e seleccionar as arestas a contrair. Estas são seleccionadas segundo critérios topológicos e geométricos. Em termos de critérios topológicos usamos os mesmos que foram aplicados por Rodrigues et. al. [Rodrigues 07], nomeadamente, num nível de simplificação se uma aresta é contraída nesse nível, todas as suas *wing edges* não podem ser contraídas nessa mesma iteração. Este critério garante a aplicação mais uniforme da simplificação, dentro do mesmo nível de simplificação e de nível para nível. Em termos geométricos, apresentamos dois critérios baseados na forma como o ser humano percepcionam os objetos, e que são:

1. Preservar as partes significativas dos objetos. Nesse sentido, a malha é segmentada uma única vez (isto é, antes de qualquer simplificação), usando o algoritmo HPO [Rodrigues 15]. Este de forma automática segmenta a malha e identifica as partes significativas dos objetos, sendo a informação dos segmentos guardada na estrutura de dados xGC. Durante a simplificação da malha, a fronteira dos segmentos não é alterada de maneira a preservar as diferentes partes do objeto. Isto vai implicar que todas as arestas que pertencem à fronteira não sejam contraídas. Nas Figuras 4, 5 e 6, pode-se verificar que a linha de fronteira entre segmentos não sofreu qualquer alteração durante as sucessivas simplificações da malha. Além disso, todas as *wing edges* dos vértices que ficam na fronteira dos segmentos (excepto as que fazem parte da fronteira), quando contraídas, terão que o ser de modo a não haver deslocação das arestas da fronteira.
2. Preservar vincos e pontos característicos. Os objetos também têm normalmente um conjunto de vincos e pontos característicos que são percecionados pelos seres humanos, o que fazemos é tentar preservá-los. Estes caracterizam-se na maior parte das vezes pelos valores elevados de curvatura [Mangan 99], sendo que uma forma de preservar esses pontos é impedir a

contração de arestas e de *wing edges* de vértices com elevada curvatura.

No seu conjunto, a segmentação da malha, os critérios de preservação de forma referidos anteriormente e a estrutura de dados xGC permitem também simplificar um conjunto específico de segmentos (simplificação parcial). Este segmentos podem ser seleccionados e simplificados de diferentes formas, como por exemplo:

1. Ponto de vista do observador. Os segmentos mais próximos do observador com simplificações menores que os que estão mais longe.
2. Seleccionado pelo utilizador. Os segmentos são seleccionados por ação direta dos utilizadores, sendo que só estes são simplificados.

No nosso algoritmo, e de forma a permitir preservar a fronteira dos segmentos, tem então que se aplicar diferentes tipos de contração de aresta (Figura 2), nomeadamente:

1. Contração para ponto médio. Quando a aresta a contrair (V_i, V_j) não pertence às *wing edges* dos vértices da fronteira, a contração desta é feita segundo a forma definida por Hoppe [Hoppe 98], ou seja, a aresta (V_i, V_j) é contraída para o ponto médio $P = (V_i + V_j)/2$ da mesma, sendo posteriormente calculado o chamado vetor de Hoppe $\vec{v}_H = V_i - P$ que permite depois reconstruir a aresta. O vértice criado é classificado como sendo do tipo 1.
2. Contração para extremidade. Quando é uma aresta que pertence às *wing edges* dos vértices da fronteira (mas não é fronteira do segmento), a aresta (V_i, V_j) é contraída para um dos pontos extremos da mesma [Kobbelt 98], isto é, $P = V_i$ se V_i é o vértice da fronteira, sendo que o vetor de Hoppe é dado por $\vec{v}_H = V_j - V_i$, e o vértice criado é classificado como sendo do tipo 2; do mesmo modo, $P = V_j$ se V_j é o vértice da fronteira, sendo que o vetor de Hoppe é dado por $\vec{v}_H = V_i - V_j$, e o vértice criado é classificado como sendo do tipo 3.

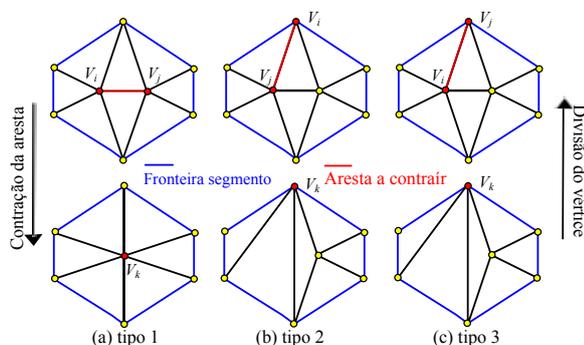


Figura 2. Tipos de contração da aresta.

O algoritmo de simplificação opera sobre o *array* de arestas de forma sequencial. Se uma aresta (V_i, V_j) é passível de ser contraída, então a contração é efectuada do seguinte modo:

1. Determina o tipo de aresta.
2. Calcula/define o novo vértice P resultante do colapso da aresta.
3. Calcula o vetor de Hoppe \vec{v}_H .
4. Testa se haverá inversão de normais das faces incidentes em V_i e V_j , se estes passarem a ter as coordenadas de P . Em caso afirmativo, é anulada a contração da aresta (situação de erro pode ser visualizada na Figura 3); caso contrário, segue-se para passos seguintes para efectivar a contração da aresta.
5. V_i é redefinido como o novo vértice V_{N+1} a partir de P , onde N indica o maior índice dos vértices.
6. Define \vec{v}_H para V_{N+1} .
7. Criar a sub-árvore genealógica de V_{N+1} , i.e. $V_{N+1}(\text{id}(V_i), \text{id}(V_j))$, sendo os identificadores $\text{id}(V_i)$ e $\text{id}(V_j)$ armazenados como nó esquerdo e direito, respetivamente. Além disso é guardada informação acerca do tipo de aresta envolvida na contração.
8. Actualiza a lista de arestas da malha.
9. Actualiza lista de arestas de V_{N+1} com as de V_j .
10. V_j guardado na lista de vértices fantasma.

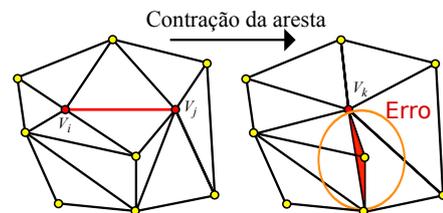


Figura 3. Erro da contração da aresta.

5 ALGORITMO DE REFINAMENTO

Este algoritmo permite restaurar a malha (ou os segmentos da malha) para o nível seguinte de maior detalhe. Basicamente o que se faz nesta iteração é aplicar o algoritmo de refinamento proposto por Rodrigues et. al. [Rodrigues 07], mas com as alterações que a seguir se descrevem.

A informação que permite restaurar a malha para um nível de detalhe superior está toda ela armazenada nas árvores genealógicas associadas aos vértices. Aquando da operação de divisão de um vértice V_k , há que primeiro verificar o tipo de contração de aresta que esteve na origem desse vértice. A partir dessa informação é possível então restaurar a geometria dos vértices que faziam parte da aresta contraída, o que é feito do seguinte modo:

1. Tipo 1. Neste caso (contração da aresta para o ponto médio), os novos vértices V_i e V_j serão iguais a $V_i = V_k - \vec{v}_H$ e $V_j = V_k + \vec{v}_H$.
2. Tipo 2. Neste caso (contração da aresta para o V_i da aresta), os novos vértices V_i e V_j serão iguais a $V_i = V_k$ e $V_j = V_k + \vec{v}_H$.
3. Tipo 3. Neste caso (contração da aresta para o V_j da aresta), os novos vértices V_i e V_j serão iguais a $V_i = V_k + \vec{v}_H$ e $V_j = V_k$.

6 RESULTADOS EXPERIMENTAIS

No final interessa conhecer quais os resultados da aplicação do esquema de multirresolução nomeadamente no que diz respeito:

1. Preservação de forma. Verificar se a forma geral do objeto e dos segmentos se mantêm com a aplicação sucessiva de simplificações. De referir que a simplificação pode ser aplicada a todo o objeto ou somente a um conjunto de segmentos da malha.
2. Taxa de simplificação. Verificar qual a percentagem de redução de polígonos da malha durante as sucessivas etapas de simplificação da malha.

Para os testes foram utilizados os modelos existentes no Princeton Benchmarking [Chen 09], que disponibiliza uma grande diversidade de objetos divididos por 19 famílias de objetos.

6.1 Preservação da forma

A fim de avaliar como a forma dos objetos é preservada, fizemos dois testes. No primeiro simplificou-se as malhas sem alterar a distância ao observador, com o intuito de demonstrar que a qualidade visual da malha se mantém. No segundo teste, a simplificação só incidiu num segmento da malha, e não na malha no seu todo. Os resultados do primeiro teste estão ilustrados nas Figuras 4 e 5, onde para dois objetos bastante diferentes se aplicaram sucessivas simplificações, nomeadamente 2 e 4 iterações. Como se pode observar, os objetos apesar de terem reduções significativas do número de faces, mantêm a forma dos mesmos e preservam a fronteira entre as diferentes regiões. Além disso, os vincos e pontos característicos dos objetos dentro de cada segmento permanecem sem alterações significativas. A junção do critério topológico com os critérios geométricos permite uma distribuição mais uniforme da simplificação, preservando a forma da malha, tanto a nível geral como em termos de segmentos e pontos característicos do objeto.

Na Figura 6 ilustra-se os resultados do segundo teste. Neste caso, pode-se observar a aplicação parcial da simplificação à asa direita da malha de um avião que foi selecionado pelo utilizador. Este segmento foi sendo sucessivamente simplificado, embora a figura só mostre a asa com 2 e 4 iterações de simplificação. Pode-se observar que a asa preserva a forma, ainda que o número de faces tenha

sido bastante reduzido, tendo passado de um número inicial de 2915 faces para 471 faces. Além disso, a fronteira do segmento da asa foi completamente preservado durante este processo.

6.2 Taxa de Simplificação

Na Figura 7 estão representados os resultados do algoritmo de simplificação em termos de percentagem de redução do número de faces, aplicados a 19 tipos de objetos diferentes. Como se pode observar, numa primeira fase existe uma redução de aproximadamente 50% no número de faces. Esta redução no número de faces dos objetos verifica-se em média até ao quinto e sexto níveis de simplificação. A partir destes níveis, a simplificação deixa de ocorrer devido aos critérios geométricos que são usados na preservação da forma dos objetos. Pode-se também constatar que as superfícies com um maior número de segmentos, com formas cilíndricas e em que o diâmetro é mais pequeno, acabam por ter no final têm uma taxa de redução de polígonos menor, pois têm uma maior curvatura.

7 CONCLUSÕES

Neste artigo foi apresentado um esquema completo para malhas segmentadas com resolução variável. Este esquema incorpora uma estrutura de dados, designada xGC, que permite armazenar malhas segmentadas e que dá suporte ao funcionamento dos algoritmos de simplificação e refinamento da malha de resolução variável que preservam a forma. A preservação da forma faz -se à custa da preservação das fronteiras entre segmentos e dos principais pontos característicos da malha. A utilização de malhas segmentadas permite variar a resolução de toda a malha ou então de só uma parte específica desta malha, isto, é de um segmento ou conjunto de segmentos.

Todo este processo pode ser feito de forma automática desde o algoritmo de segmentação ao algoritmo de simplificação, pois o método de simplificação pára automaticamente na quinta ou na sexta iteração de simplificação, em resultado dos critérios geométricos impostos no processo de simplificação.

Como trabalho futuro, pretende-se fazer a introdução de uma suavização da malha após cada simplificação, mas esta operação implica alterações a nível do armazenamento da informação na genealogia dos vértices.

Referências

- [Attene 06] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, e A. Tal. Mesh segmentation — a comparative study. Em *Proceedings of the IEEE International Conference on Shape Modeling and Applications (SMI'2006)*, páginas 7–19. IEEE Computer Society, 2006.
- [Biederman 87] Irving Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147, 1987.

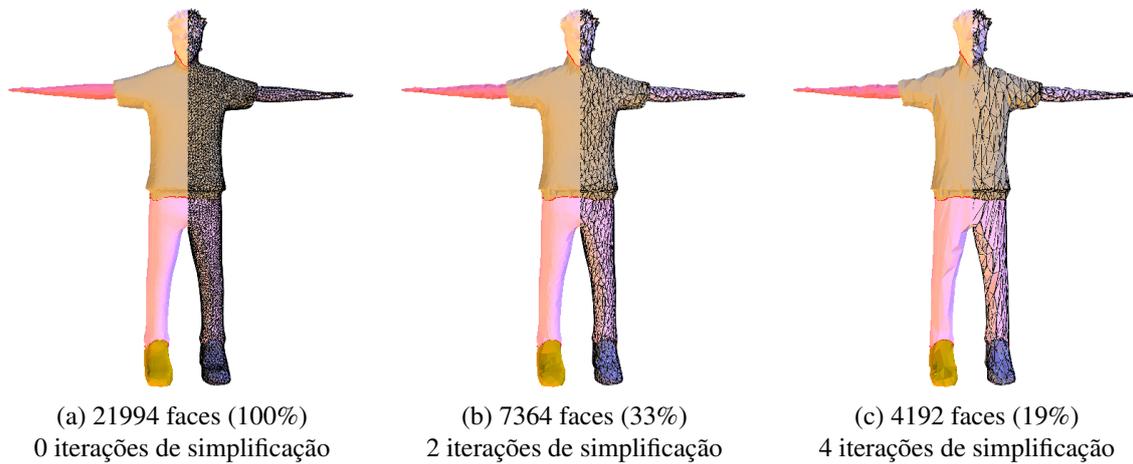


Figura 4. Simplificação de malha de ser humano .

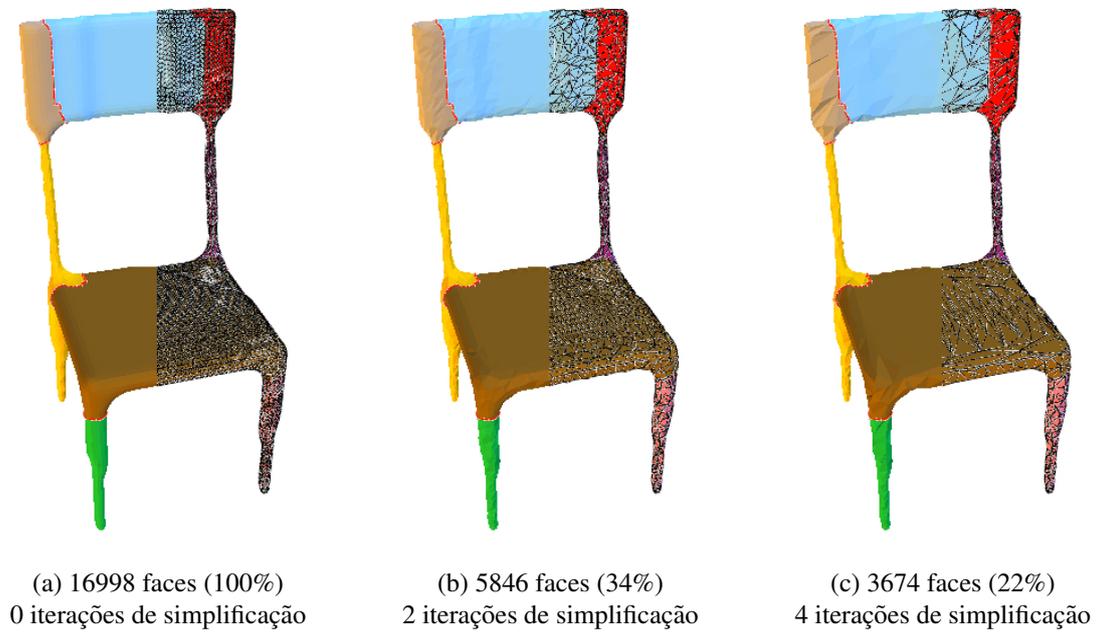


Figura 5. Simplificação de malha de uma cadeira .

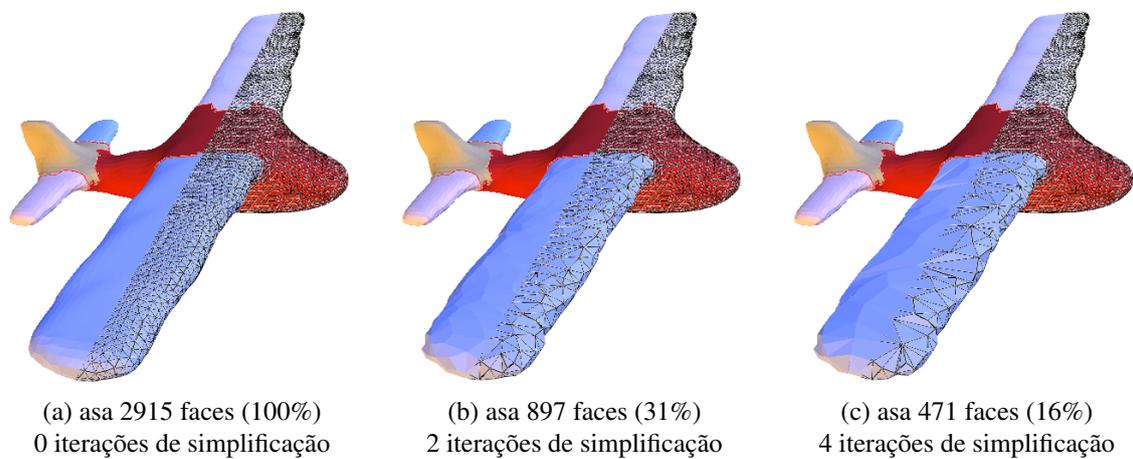


Figura 6. Simplificação da asa direita da malha de um avião .

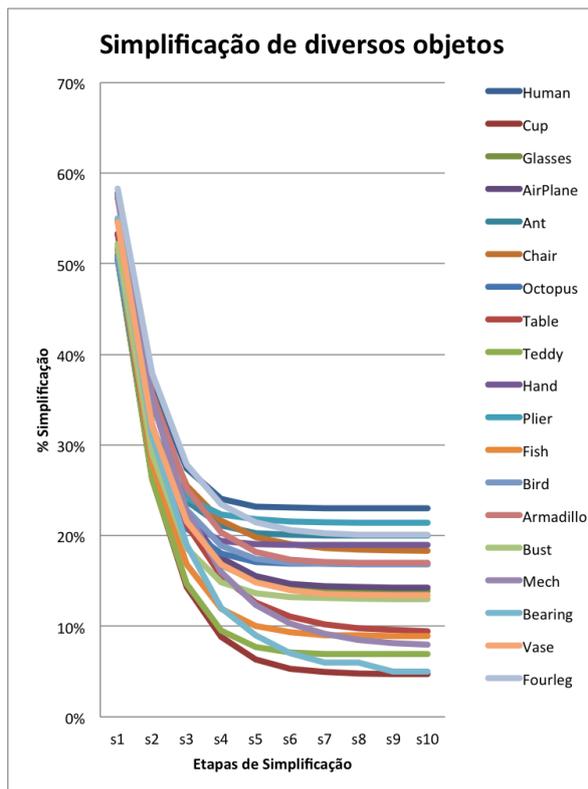


Figura 7. Evolução da simplificação da malha.

- [Chen 09] Xiaobai Chen, Aleksey Golovinskiy, e Thomas Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics*, 28(3):73:1–73:12, August 2009.
- [Cheng 07] Zhi-Quan Cheng, Hua-Feng Liu, e Shi-Yao Jin. The progressive mesh compression based on meaningful segmentation. *The Visual Computer*, 23(9-11):651–660, 2007.
- [Choe 09] Sungyul Choe, Junho Kim, Haeyoung Lee, e Seungyong Lee. Random accessible mesh compression using mesh chartification. *IEEE Transactions on Visualization and Computer Graphics*, 15(1):160–173, Janeiro 2009.
- [Ciampalini 96] A. Ciampalini, P. Cignoni, C. Montani, e R. Scopigno. Multiresolution decimation based on global error. Relatório técnico, Paris, France, France, 1996.
- [Danovaro 05] E. Danovaro, L. Floriani, P. Magillo, E. Puppo, D. Sobrero, e N. Sokolovsky. The half-edge tree: A compact data structure for level-of-detail tetrahedral meshes. Em *SMI '05: Proceedings of the International Conference on Shape Modeling and Applications 2005 (SMI'05)*, páginas 334–339, Washington, DC, USA, 2005. IEEE Computer Society.
- [ES99] Jihad El-Sana e Amitabh Varshney. Generalized view-dependent simplification. *Computer Graphics Forum*, 18(3):83–94, 1999.
- [Floriani 02] Leila Floriani e Paola Magillo. Multi-resolution mesh representation: Models and data structures. Em *Tutorials on Multiresolution in Geometric Modelling*, páginas 363–417. Springer, 2002.
- [Garland 97] Michael Garland e Paul S Heckbert. Surface simplification using quadric error metrics. Em *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, páginas 209–216. ACM Press/Addison-Wesley Publishing Co., 1997.
- [Garland 99] Michael Garland. Multiresolution modeling: Survey & future opportunities. *State of the Art Report*, páginas 111–131, 1999.
- [Hoffman 97] Donald D. Hoffman e Manish Singh. Saliency of visual parts. *Cognition*, 63(1):29 – 78, 1997.
- [Hoppe 93] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, e W. Stuetzle. Mesh optimization. Em *SIGGRAPH '93: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, páginas 19–26, New York, USA, 1993. ACM Press.
- [Hoppe 96] Hugues Hoppe. Progressive meshes. Em *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, páginas 99–108. ACM, 1996.
- [Hoppe 97] Hugues Hoppe. View-dependent refinement of progressive meshes. view-dependent refinement of progressive meshes. Em *Proceedings of the 24rd annual conference on Computer graphics and interactive techniques*, volume 31, páginas 189–198. ACM, 1997.
- [Hoppe 98] H. Hoppe. Smooth view-dependent level-of-detail control and its application to terrain rendering. Em *VIS '98: Proceedings of the Conference on Visualization '98*, páginas 35–42, Los Alamitos, CA, USA, 1998. IEEE Computer Society Press.

- [Kim 01] Junho Kim e Seungyong Lee. Truly selective refinement of progressive meshes. Em *Graphics Interface*, volume 1, páginas 101–110, 2001.
- [Kim 06] Junho Kim, Sungyul Choe, e Seungyong Lee. Multiresolution random accessible mesh compression. Em *Computer Graphics Forum*, volume 25, páginas 323–331. Wiley Online Library, 2006.
- [Kobbelt 98] Leif Kobbelt, Swen Campagna, e Hans-Peter Seidel. A general framework for mesh decimation. Em *Graphics interface*, volume 98, páginas 43–50, 1998.
- [Luebke 97] D. Luebke e C. Erikson. View-dependent simplification of arbitrary polygonal environments. Em *SIGGRAPH '97: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, páginas 199–208, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.
- [Maglo 11] Adrien Maglo, Ian Grimstead, e Céline Hudelot. Cluster-based random accessible and progressive lossless compression of colored triangular meshes for interactive visualization. Em *Computer Graphics International*, volume 11, 2011.
- [Maglo 13] Adrien Maglo, Ian Grimstead, e Céline Hudelot. Pomar: Compression of progressive oriented meshes accessible randomly. *Computers and Graphics*, 37(6):743 – 752, 2013. Shape Modeling International (SMI) Conference 2013.
- [Mangan 99] Alan Mangan e Ross Whitaker. Partitioning 3D surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):308–321, October 1999.
- [Ni 94] Xiujun Ni e M Susan Bloor. Performance evaluation of boundary data structures. *Computer Graphics and Applications, IEEE*, 14(6):66–77, 1994.
- [Pajarola 01] R. Pajarola. Fastmesh: Efficient view-dependent meshing. Em *PG '01: Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*, página 22, Washington, DC, USA, 2001. IEEE Computer Society.
- [Rodrigues 07] Rui Rodrigues, José Morgado, Frutuoso Silva, e Abel Gomes. A ghost cell-based data structure for multiresolution meshes. Em *Computational Science and Its Applications–ICCSA 2007*, páginas 666–679. Springer, 2007.
- [Rodrigues 15] Rui S.V. Rodrigues, José F.M. Morgado, e Abel J.P. Gomes. A contour-based segmentation algorithm for triangle meshes in 3d space. *Computers & Graphics*, 49(0):24 – 35, 2015.
- [Rossignac 93] J. Rossignac e P. Borrel. Multi-resolution 3D approximations for rendering complex scenes. Em B. Falcidieno e T. Kunii, editores, *Modeling in Computer Graphics: Methods and Applications*, Berlin, 1993. Springer-Verlag.
- [Schröder 00] Peter Schröder, Denis Zorin, T DeRose, DR Forsey, L Kobbelt, M Lounsbery, e J Peters. Subdivision for modeling and animation. *SIGGRAPH 2000 Course Notes*, 2000.
- [Schroeder 92] W. J. Schroeder, J. A. Zarge, e W. E. Lorenson. Decimation of triangle meshes. Em *SIGGRAPH '92: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*, páginas 65–70, New York, NY, USA, 1992. ACM Press.
- [Shamir 08] Ariel Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, September 2008.
- [Silva 03] Frutuoso GM Silva e Abel JP Gomes. Adjacency and incidence framework: a data structure for efficient and fast management of multiresolution meshes. Em *Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, páginas 159–166. ACM, 2003.
- [Silva 04] Frutuoso GM Silva e Abel JP Gomes. Normal-based simplification algorithm for meshes. Em *Theory and Practice of Computer Graphics, 2004. Proceedings*, páginas 211–218. IEEE, 2004.
- [Yang 04] Sheng Yang, Chang-Su Kim, e CC Jay Kuo. A progressive view-dependent technique for interactive 3-d mesh transmission. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(11):1249–1264, 2004.

Uma aplicação móvel para implementar o conceito de caderneta de cromos aos Museus

Resumo

No âmbito do património cultural, os museus são uma peça central. No entanto, carecem de atualização face às necessidades de experiências mais interativas, de forma a cativar um público mais jovem. Tendo como caso de estudo os museus com uma vertente marítima, é objetivo do presente trabalho apresentar uma aplicação móvel com Realidade Aumentada (Aplicação Museu C+) que permite descobrir os barcos e as suas particularidades de forma mais interativa e envolvente. Para tal recorre-se a uma abordagem gamificada sobre um conceito de caderneta de cromos, e integrando realidade aumentada. Uma das questões de investigação solucionada neste trabalho é como integrar nos museus a componente tecnológica numa perspetiva de colecionismo digital. Antes de definir a arquitetura da aplicação foi realizada uma pesquisa de forma a identificar os requisitos mais relevantes a introduzir. Neste sentido, é projetada uma caderneta de cromos digital assente em princípios inerentes à gamificação. A mecânica base é o colecionismo, tendo sido desenvolvida uma aplicação móvel que possibilita adquirir um cromo quando se identifica um determinado elemento de um barco, diretamente através de uma funcionalidade de realidade aumentada. Assim, os cromos digitais obtêm-se ao ultrapassar desafios no contexto do museu, despoletando uma conjugação de elementos de realidade mista sob a forma de modelação tridimensional, texto e vídeos. Após a experiência concreta no museu, a caderneta de cromos digital permitirá desbloquear os modelos tridimensionais dos barcos cujos cromos foram completados, servindo assim ela própria como uma recompensa tangível, um souvenir. Na avaliação do protótipo desenvolvido, os resultados apontam para o manifesto interesse da aplicação.

Palavras-chave

Colecionismo, gamificação, museus, realidade aumentada, realidade virtual, aplicações móveis.

1. INTRODUÇÃO

Os museus consolidam o património de um país e de uma dada região, ao interligar a história, a arte, a ciência, o território e o mar, entre outros. A confirmar a sua importância observa-se um acréscimo gradual do número de visitantes segundo dados mais recentes do Instituto Nacional de Estatística [INE14]. No entanto, a revisão de literatura e inquérito preliminar, apresentado no âmbito de um estudo prévio [BFRP], apontam para uma lacuna nos meios interativos disponíveis nos museus para acompanhar a recente evolução tecnológica, de forma a efetivamente atrair um público mais jovem.

Partindo das embarcações típicas portuguesas, como um marco inalienável da história de Portugal, presentes em museus com uma vertente marítima, propõe-se no presente artigo uma aplicação móvel, denominada “Museu C+”, que possibilite descobrir os barcos e as suas particularidades de forma mais aliciante. Esta proposta, embora focada para a exploração das embarcações típicas de um museu marítimo, poderá ser generalizada para outro tipo de museus. Para o efeito encontra-se subjacente uma abordagem gamificada sobre um conceito de caderneta de cromos, integrando realidade aumentada.

A principal questão científica patente neste trabalho era como integrar os museus e a componente tecnológica numa perspetiva de colecionismo digital? O colecionismo é uma atividade patente em muitas pessoas e identifica-se claramente com a maior parte dos museus.

A solução desenvolvida consubstancia-se numa caderneta de cromos digital assente em princípios inerentes à gamificação. A mecânica base é o colecionismo, tendo sido concebida uma aplicação que possibilita obter um cromo quando se identifica um determinado elemento de um barco, diretamente através de uma funcionalidade de realidade aumentada. Assim, os cromos digitais resultam como recompensa de ultrapassar desafios no contexto do museu, despoletando uma junção de elementos sob a forma de modelação tridimensional, texto e vídeos. Após a experiência concreta no museu, a caderneta de cromos digital permitirá desbloquear os modelos tridimensionais dos barcos cujos cromos foram completados, tornando-se ela própria como uma recompensa tangível, um souvenir.

2. TRABALHO RELACIONADO

O trabalho relacionado com o presente estudo compreende aplicações tecnológicas disponíveis em museus, bem como os conceitos de embarcações, colecionismo, gamificação e realidade aumentada.

Na esfera museológica, tem-se vindo a assistir nos últimos anos à introdução de soluções tecnológicas sob diversos suportes, com vista a melhorar a experiência dos visitantes. Por um lado, os sistemas para visita auto-guiada têm vindo a incorporar elementos multimédia [CCH14] [RRC*13]. Por outro, incluem-se tecnologias de realidade mista para recriar peças tridimensionais [Cho14] [JNRR15] [KPR*13], enfoque em pormenores do espólio museológico [CFQP14] [Cho14] [KPR*13] e recriação de exposições virtuais [JP12] [RRC*13], entre

outros [CC13] [CPZS13]. Em suma, a literatura confirma o potencial deste tipo de soluções [Cho14] [TTK*13] [YW14]. No entanto, duas lacunas ressaltam dos anteriores trabalhos, que se relacionam com a informação multimédia apenas estar disponível no momento da experiência, sem com isso incrementar a interatividade com o utilizador.

As embarcações típicas têm vindo a ocupar um papel de relevo nos últimos anos, como parte integrante do património cultural português, onde se assiste a um investimento significativo de retratar a sua história, desde as especificações técnicas à respetiva caracterização [Fon15] [Fon11]. Mas, atualmente as mesmas surgem expostas de forma passiva sem induzir a interatividade com o utilizador ou a estimulação cognitiva.

O conceito de colecionismo, enraizado na caderneta de cromos, este pode ser definido como o processo de adquirir e possuir coisas de forma ativa, seletiva e apaixonada [Bel95]. Distintos públicos-alvo têm vindo a eternizar o colecionismo, com os cromos de futebol e personagens do mundo do fantástico, ao universo do adulto com objetos colecionáveis e estratégias de marketing a eles direcionadas. Mas, decisivamente o suporte físico tem-se sobrepuesto ao digital.

Numa abordagem de gamificação, esta surge associada à utilização de elementos de jogo e técnicas de Game Design a contextos de não jogo, tendo recentemente emergido com uma base científica consolidada [Arm13] [DDKN11] [Jag13] [KL12] [Mat12] [SRV13], apesar de não existir uma definição universal [Fer13] [SF14]. Atualmente uma multiplicidade de áreas têm incorporado os princípios de gamificação, desde o mundo empresarial ao ambiente académico [Jag13] [ZL10], e resultados benéficos têm vindo a ser alcançados [Arm13] [Jag13] [KL12] [SF14]. Na implementação da estratégia de gamificação, diferentes elementos surgem incorporados sob a forma de mecanismos e dinâmicas [BL13] [Nee12] [SRV13]. Por um lado, os mecanismos correspondem a regras onde se inclui o comportamento, sistema de pontuação, troféus, objetos virtuais, níveis, tarefas, quizzes, avatars, etc. Por outro, as dinâmicas almejam tornar o jogo/tarefa agradável, incorporando comportamentos de coleção, exploração, cooperação, competição, organização e desafios. Em suma, pretende-se motivar a estimulação cognitiva e curiosidade intelectual, em simultâneo com a recompensa e o reconhecimento.

Por fim, relativamente às tecnologias com potencialidades de serem introduzidas no espaço museológico sem incluir intrusão no mesmo, importa salientar a tendência crescente de combinar ambientes reais e virtuais, sob um espectro de realidade mista [BCL14] [MK94]. Por realidade mista, entende-se um cenário híbrido onde o mundo real e os objetos virtuais são visualizados conjuntamente. Quanto à realidade virtual (RV), esta compreende um cenário onde o participante/observador está integralmente submerso num mundo artificial, com características reais clonáveis e obstáculos ultrapassados. Relativamente à realidade aumentada (RA) esta corresponde a uma sobreposição de objetos virtuais no mundo real, com poten-

cialidades promissoras [CKW13] [LGG13] [RMSN12]. As respetivas aplicações têm vindo a ser acolhidas nas mais diversas áreas, tais como a educação, a cultura, a defesa nacional, a saúde, o entretenimento, o marketing, entre muitos outros domínios de atuação [BKP01] [CCH*12] [CKW13] [CPG13] [Goe13] [JQCVB10] [LCYL06] [LGF12] [NOCM12] [RKW13] [RMSN12] [SRV13].

3. ARQUITETURA DA APLICAÇÃO MUSEU C+

Previamente à definição da arquitetura da solução foi realizada uma pesquisa, sob a forma de um inquérito preliminar, de modo a identificar os requisitos mais relevantes a introduzir numa aplicação [BFRP]. Numa amostra de 47 inquiridos, as conclusões apontam para a importância de visitar museus, ocorrendo a visita no seio da dinâmica familiar, sendo que os que não visitam apontam como justificação a falta de interesse e a reduzida interatividade tecnológica. Uma parte substancial da amostra atualmente coleciona algo, ou pelo menos já o fez no passado. Quando se introduziu um estudo de caso com as possíveis componentes (realidade aumentada, virtual, animação tridimensional, texto, vídeos e caderneta de cromos) a introduzir numa aplicação em espaço museológico, a realidade aumentada surge como um dos elementos mais preponderante, seguindo-se com idêntico grau de importância a realidade virtual, animação tridimensional e vídeos.

Passando para a arquitetura da aplicação a desenvolver, a Figura 1 esquematiza sequencialmente as etapas da solução e apresenta os elementos de gamificação envolvidos, que serão posteriormente detalhados.

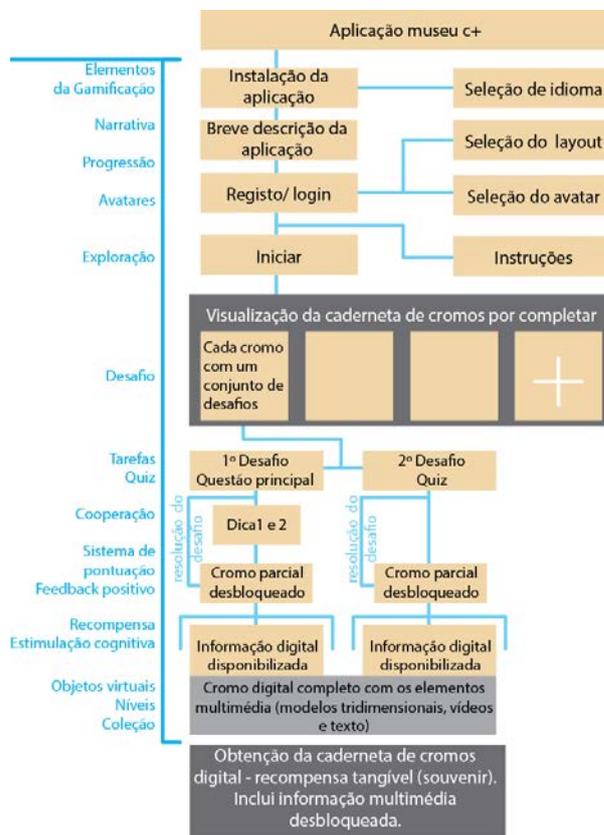


Figura 1 Esquema da solução desenvolvida

Assim, na recepção, o visitante é convidado a conhecer a aplicação Museu C+, para decidir se entra no desafio. Dado o *feedback* obtido no inquérito preliminar, entende-se que, pelo facto das visitas ocorrerem sobretudo em família, esta possa suscitar maior interesse na conjugação de conhecimentos.

Logo que seja feito *download* da aplicação, e após seleccionar-se o idioma, surge uma breve introdução sobre qual é o seu objetivo. Posteriormente surge o menu de registo / login, que poderá ser feito através do Google ou Facebook tornando assim possível publicitar a aplicação nestas redes sociais, numa perspetiva de transmedia. Após introdução dos dados, a fase seguinte consiste na escolha do tema com que o jogo surge desenhado: versão infantil ou histórica. No primeiro caso, as cores surgem mais vivas, a linguagem mais simples, e os conteúdos adaptados ao público infantil (Figura 2). Na componente histórica, o *layout* surge adaptado à cartografia histórica e com cores alusivas à época dos descobrimentos em tons de castanho, com elementos fulcrais em tons de azul a simbolizar o mar e a ria (Figura 3). Por simplificação, apenas o *layout* da versão histórica será aqui explorado. Numa fase seguinte surge a opção de seleccionar o avatar para jogar, inspirado em trajes típicos, que tanto podem ser adaptados ao género feminino – a varina, como ao género masculino – o típico pescador ou marnoto (Figura 4). Subsequentemente, é mostrada a caderneta de cromos digital por preencher, e uma breve nota introdutória.



Figura 2 Caderneta - versão infantil



Figura 3 Caderneta - versão histórica



Figura 4 Seleção do avatar

Para desbloquear a caderneta de cromos digital, é proposto um conjunto de desafios adaptado a cada embarcação. Assim, no primeiro desafio começa por surgir uma questão principal, apoiada por dois botões de ajuda, com pistas de teor diferente. Para estes desafios é necessário utilizar o modo de RA, no qual, após apontar para a peça física correta da embarcação, despoletará informação referente à utilidade da peça. Após ultrapassar a primeira parte do desafio, é desbloqueado o lado esquerdo do cromo. Neste caso, surge uma breve descrição do barco, e uma imagem elucidativa que ajudará a responder à segunda questão do desafio, e que permitirá desbloquear o lado direito do cromo digital. Nesta nova pergunta, o grau de dificuldade é substancialmente reduzido consistindo numa escolha múltipla sobre duas opções. Após seleccionar a opção correta, o cromo digital surge completo e novos conteúdos multimédia disponíveis, tais como planos tridimensionais da embarcação e uma pequena animação de como se constrói a embarcação. Assim, o utilizador passa de nível após ultrapassar cada desafio, ganhando acesso a um cromo digital (e conteúdo multimédia associado) como recompensa tornando mais cativante a exploração do museu e a apreensão real de conhecimento.

Uma vez que se pretende promover a aprendizagem, os desafios conseguem ser sempre superados com as ajudas fornecidas. De notar que se encontra disponível a visualização dos conteúdos conquistados. Quase sem se aperceber o utilizador encontra-se a jogar num contexto real de museu, em paralelo com a aprendizagem de novos conhecimentos, numa experiência interativa. E neste caso o participante / jogador sente que é a peça central do jogo. A progressão do jogo ocorre de forma intuitiva. A descoberta do museu para a coleção dos cromos torna a experiência mais interessante, ao mesmo tempo que se pretende promover a cooperação entre grupos de amigos ou família. Ao longo dos cinco desafios, diferentes níveis vão sendo ultrapassados e a pontuação é atribuída mediante o desbloqueio de cada cromo. No final da solução, os cromos digitais estão à disposição do utilizador, juntamente com a pontuação e um crachá de reconhecimento pelo esforço, em que lhe confere o grau de comandante da embarcação.

No global, diferentes elementos de gamificação vão estando presentes. No que respeita às mecânicas incluem-se um sistema de pontuação, crachás, níveis, tarefas, quizzes, avatares e objetos virtuais. Relativamente às dinâmicas, estas residem na coleção, na exploração, na coopera-

ção e no desafio. Ao nível dos motivos salienta-se a curiosidade intelectual, estimulação cognitiva, reconhecimento e recompensa.

4. APLICAÇÃO MUSEU C+

Museu C+ é a designação da aplicação, que acopla termos que traduzem a vertente de aplicação ao museu, bem como a obtenção de uma caderneta de cromos digital associada ao objetivo de mais conhecimento.

Esta secção apresenta os *layouts* da solução móvel desenvolvida, seguindo a arquitetura previamente detalhada. Assim, numa primeira fase surge a escolha do idioma apresentando-se de seguida uma breve nota introdutória como vista inicial da aplicação (Figura 5).



Figura 5 Vista inicial

Posteriormente um conjunto de desafios é proposto a ser ultrapassado, por forma a conseguir desbloquear a caderneta de cromos digital (Figura 6).

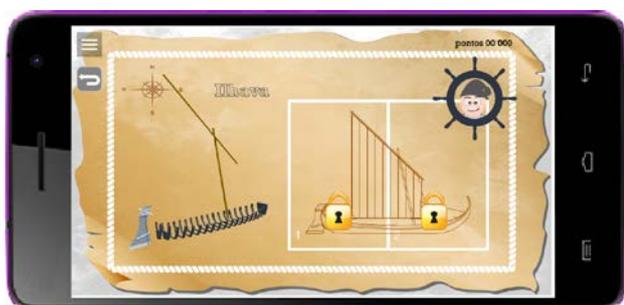


Figura 6 Cromo a desbloquear em 2 fases

Ao nível dos desafios propostos, observa-se na Figura 7 a questão principal, seguida de dois botões de ajuda a poderem ser utilizados.



Figura 7 Desafio – questão principal com ícones ajuda

A vertente final inclui a atribuição de um crachá digital, como sinal de reconhecimento, e da caderneta de cromos digital com todos os conteúdos desbloqueados como recompensa pelos desafios ultrapassados (Figura 8).



Figura 8 Etapa final

4.1 Módulo de realidade aumentada

Num estágio inicial procedeu-se a um teste na ferramenta Artoolkit com o objetivo de avaliar os modelos das embarcações na vertente de realidade aumentada. Pretendia-se pois aferir o comportamento do ficheiro de modelação tridimensional em suporte da tecnologia de realidade aumentada (Figura 9).



Figura 9 Modelação tridimensional em suporte RA

Posteriormente avaliou-se o referido modelo tridimensional com efeitos de animação, e os resultados corroboraram o que se pretendia implementar. Este despoletar de realidade aumentada iniciou-se através de fiduciais normais (Figura 10), evoluindo de forma subsequente para fiduciais personalizados (Figura 11) indo ao encontro de todo o grafismo da aplicação a desenvolver.

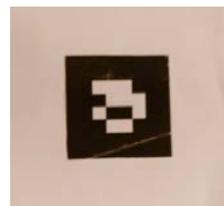


Figura 10 Fiducial normal

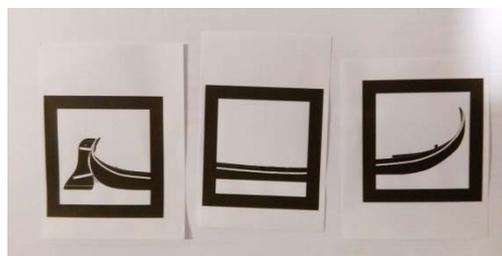


Figura 11 Fiducial personalizado

Após esta validação inicial, seguiu-se todo o desenvolvimento gráfico da aplicação, dadas as respostas favoráveis obtidas nesta fase.

5. DESENVOLVIMENTO DA APLICAÇÃO

A aplicação é projetada por um lado para se conjugar com embarcações físicas reais e, por outro, para dar uma

nova dimensão de realidade a imagens estáticas que possam ser o único elemento em exposição. Tomando em linha de conta a relevância histórica de embarcações tradicionais, foram estudadas em detalhe cinco embarcações para dar desenvolver: Moliceiro, Mercantel, Ladra, Ílhava e Barco do Mar.

De modo a apresentar a perspetiva de utilização da aplicação, a Figura 12 detalha o conjunto de desafios que se encontram delineados para uma embarcação específica: a Ílhava. Para esta embarcação específica, a questão central consiste em “Encontre o elemento coicía”. Neste caso as ajudas disponíveis a despoletar correspondem a: (1) encontra-se na parte central da embarcação; (2) tem um determinado formato (desenho). Esta questão e respetivas ajudas direcionam o visitante a tomar conhecimento de uma parte fundamental da embarcação que segura o mastro, sendo este retirado em caso de intempérie. Desta forma, pretende-se também evidenciar as condições adversas a que poderiam estar sujeitos os tripulantes. O quiz que permitirá desbloquear a parte remanescente do cromo digital consiste em responder à seguinte escolha múltipla: “De onde é a Ílhava?” A resposta correta neste quiz é Ílhavo. Neste caso, torna-se direta a associação do nome da embarcação.

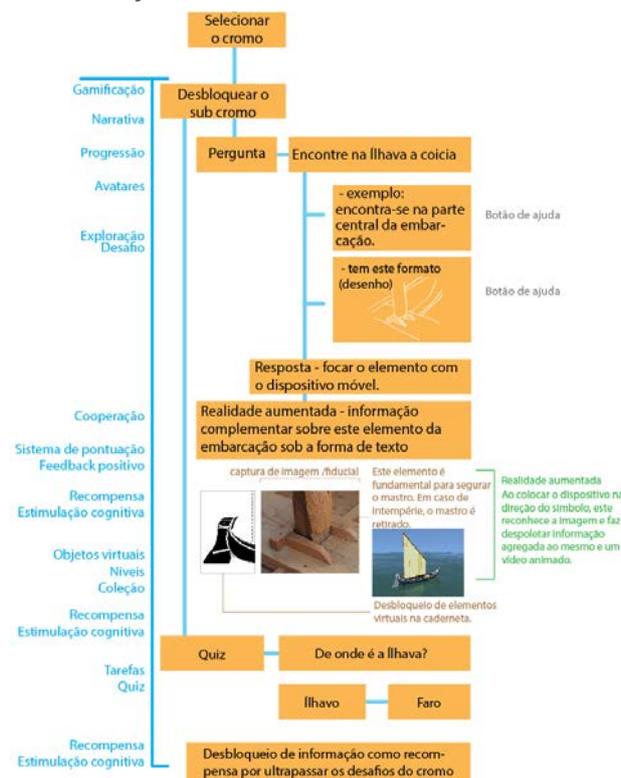


Figura 12 Esquematisação dos desafios para a Ílhava

Para desbloquear um cromo digital subsequente, os novos desafios centram-se no Moliceiro. Para esta peça museológica surge o seguinte desafio: “Encontre no Moliceiro a porta do castelo”. Neste caso as ajudas disponíveis a despoletar correspondem a: (1) encontra-se na parte frontal da embarcação; (2) tem um determinado formato (desenho). O quiz que permitirá desbloquear a parte remanescente do cromo digital consiste em responder à seguinte escolha múltipla: “O que se guardava na porta do caste-

lo” A resposta correta neste quiz é comida. O moliceiro é sobejamente reconhecido pela sua decoração, em que por norma é dada especial atenção pelos visitantes. Desta forma, pretende-se também reforçar a ideia da parte funcional, para além da finalidade a que se destina e que neste caso surgirá sob a forma de texto.

Estrategicamente posicionado a seguir ao Moliceiro, o Mercantel também tem uma decoração com traços populares que merecem ser apreciados com maior detalhe, e por isso o enfoque que aqui é dado. Assim, a questão central consiste em “Encontre no Mercantel a flor da ré”. Neste caso as ajudas disponíveis a despoletar correspondem a: (1) encontra-se na parte posterior da embarcação; (2) tem um determinado formato (desenho). O quiz corresponde a “O que pode substituir a vela?”, sendo motor a resposta correta, para que os utilizadores possam interligar o passado ao presente, ou seja, a vela de outrora é atualmente substituída pelo motor que conduz a maior velocidade, menos esforço humano, e mais resistente às condições climáticas.

Para a Ladra é proposto o seguinte desafio. Como questão central surge “Encontre na Ladra os forcados da ré”. As ajudas a despoletar correspondem a: (1) encontra-se na parte posterior da embarcação; (2) tem um determinado formato (desenho). O quiz que permitirá desbloquear a parte remanescente do cromo digital consiste em responder à seguinte escolha múltipla: “Quantas cavernas tem a Ladra?” A resposta correta neste quiz é o número 6. Neste caso a questão principal e o quiz estão intimamente relacionados no que respeita à estrutura vital que suporta a embarcação, e acessível à visualização direta por parte do visitante. A simplicidade dos pormenores técnicos desta embarcação, com um nome sui generis, torna os termos mais facilmente apreendidos.

Por fim, o Barco do Mar assume a inexistência da peça museológica, e prossegue com a seguinte indicação: “Desloque-se ao Piso 2 e encontre no mural das embarcações do Barco do Mar”. Este é um convite que indiretamente conduz o visitante a prestar atenção redobrada a pormenores que se encontram expostos ao longo da nova sala a descobrir. Neste caso as ajudas disponíveis a despoletar correspondem a: (1) encontra-se entre a Neta da Nazaré e a Barca da Serrana; (2) está no moral das embarcações. Neste caso, o quiz que permitirá desbloquear a parte remanescente do cromo digital reforça o papel da cultura marítima de cada região, onde pretende resposta à seguinte escolha múltipla: “De onde é característico?” No caso de visitantes nacionais, oriundos de diferentes pontos do país, têm à sua disposição informação global de embarcações que lhes podem ser particularmente familiares dada a proximidade.

6. AVALIAÇÃO PRELIMINAR

Tendo por base a aplicação desenvolvida, ainda em fase de protótipo, foi realizado uma avaliação preliminar com especialistas, através de um questionário aplicado após uma sessão de experimentação com aplicação. Com base na revisão de literatura [SCT*14], entendeu-se utilizar como métricas de avaliação as seguintes: atenção, relevância, satisfação, conformidade com a perceção do utili-

zador, usabilidade, efetividade, utilidade, facilidade de uso de acordo com instruções fornecidas, interesse, adequabilidade ao objetivo, etc.

Por forma a ser validado o protótipo da solução apresentada, mas assegurando uma apresentação pública restrita nesta fase inicial, foi entendido como relevante aplicar o questionário no museu marítimo para o qual foi desenvolvido o protótipo. O foco do questionário centrou-se em quinze inquiridos chave, desde o Diretor do próprio museu, passando por funcionários e visitantes do mesmo, bem como um dos autores de diversos livros de embarcações típicas. As questões colocadas ao nível dos diversos parâmetros a avaliar implicavam selecionar de 1 a 5 a opinião do inquirido, onde 1 correspondia a insuficiente e 5 a excelente. Na Figura 13 sintetiza-se graficamente a média e desvio padrão das respostas obtidas, de acordo com as seguintes métricas chave: *layout* gráfico, relevância dos conteúdos e da aplicação, interatividade, feedback ao utilizador, usabilidade, efetividade, adequabilidade das instruções, objetivos e aprendizagem, bem como o grau de satisfação com a aplicação.

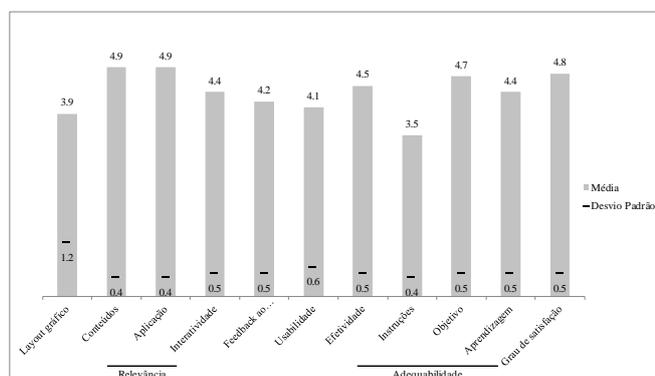


Figura 13 Análise gráfica do questionário referente à validação do protótipo

Os resultados apontam para uma melhoria a introduzir ao nível das instruções, bem como inclusão nos layouts gráficos de elementos alusivos à Ria. De forma global, é atribuída uma pontuação de excelente à relevância da aplicação e dos conteúdos. No campo de comentários, como opção colocada para melhor aferir a opinião dos inquiridos, é também possível aferir um grau de satisfação elevado com a introdução desta aplicação no Museu, o que permite antever o potencial da mesma.

7. CONCLUSÕES E TRABALHO FUTURO

Neste artigo apresenta-se o processo de conceção e desenvolvimento de uma aplicação móvel para museus, com base no conceito de caderneta de cromos digital. Por ter sido identificada uma lacuna de interatividade tecnológica no património cultural por excelência, o próximo passo foi aliar princípios de gamificação sob o suporte da realidade mista a embarcações típicas. Para o efeito, o trabalho partiu de um inquérito preliminar para avaliar as características relevantes da aplicação a desenvolver. Após o desenvolvimento do protótipo inicial, os resultados foram avaliados em contexto prático, e após essa análise melhorias serão introduzidas. De forma inequívoca os resultados comprovam o interesse na aplicação.

Relativamente à questão de investigação inicialmente formulada, observa-se que a combinação de realidade aumentada e virtual permite apresentar de forma interativa e inovadora novos conteúdos multimédia no espaço museológico, onde as embarcações tradicionais se inserem. Ao nível do despoletar de realidade mista, a solução compreende como leitura de marcador os próprios elementos da embarcação não se adicionando assim qualquer intrusão secundária no espólio museológico. A abordagem de gamificação em que se baseia a solução parte de uma narrativa em torno da descoberta de embarcações típicas onde um conjunto de desafios vai sendo ultrapassado, e feedback positivo vai gradualmente sendo atribuído, culminando com a recompensa última de uma caderneta de cromos digital que se torna possível de partilhar nas redes sociais. Por fim, a integração dos museus com a componente tecnológica numa perspetiva de colecionismo traduz-se na conquista de cromos digitais, cada um representando uma embarcação, e que através da superação de vários desafios permite alcançar uma caderneta de cromos digital.

As principais contribuições deste trabalho unificam uma perspetiva académica e prática. Por um lado, propõe-se uma nova abordagem baseada na interligação da gamificação com tecnologias de realidade aumentada e virtual e com o suporte de conteúdos multimédia. Esta abordagem baseia-se no conceito de uma caderneta de cromos digital, onde a motivação surge na égide do comportamento do colecionismo. Por outro, através de uma nova aplicação móvel pretende-se proporcionar uma nova solução tecnológica para a visita a museus, promovendo uma combinação apelativa e inovadora para a exploração das embarcações típicas portuguesas, mas que poderá ser alargado a outro tipo de museus. Neste momento há interesse por parte de um Museu Marítimo em prosseguir com a implementação da solução. Os modelos e desenhos das embarcações estão publicados num livro [Fon15].

Encontra-se em desenvolvimento a aplicação móvel, em Unity 3D, incorporado o software Qualcomm Vuforia. Esta biblioteca de RA, para além de permitir desenvolver aplicações de RA em dispositivos móveis com sistema Android e iOS, tem como vantagem adicional o uso de marcadores mais complexos, que se tornam menos sensíveis ao movimento e as imagens ficam mais estáveis [RRC*13]. Numa perspetiva de trabalho futuro, a definição de uma caderneta digital induz uma importância acrescida quando pensamos não só nos museus marítimos, mas também no panorama museológico português, o que se poderá refletir nas bases para a criação de uma caderneta de cromos digital aplicada aos museus de Portugal. O colecionismo aliado à gamificação e às novas tecnologias dão assim uma vida renovada à experiência em contexto de museus, com o valor acrescido da obtenção de recompensas tangíveis que podem ser levadas como souvenir. Posteriormente, pretende-se aplicar uma abordagem transmedia, incorporando a narrativa museológica através de outros media, tais como conteúdos multimédia e bibliografia em web sites de referência e nas redes sociais. A aplicação poderá servir assim de base para divulgar novos eventos e exposições. Por fim, esta

aplicação poderá estar ligada a um livro digital pronto a ser impresso e compilar diferentes momentos da experiência museológica do utilizador.

8. BIBLIOGRAFIA

- [Arm13] Armstrong, D.: The new engagement game: the role of gamification in scholarly publishing. *Learned Publishing*, 26 (2013), 253-256.
- [Bel95] Belk, R.: Collecting as Luxury Consumption: Effects of Individuals and Households. *Journal of Economic Psychology*, 16(3) (1995), 477-492.
- [BCL14] Billinghamurst, M., Clark, A., & Lee, G.: A survey of augmented reality. *Foundations and Trends in Human-Computer Interaction*, 8 (2014), 73-272.
- [BKP01] Billinghamurst, M., Kato, H., & Poupyrev, I.: The MagicBook: a transitional AR interface. *Computers & Graphics* 25 (2001), 745-753.
- [BL13] Blohm, I., & Leimeister, J.M.: Gamification. Design of ITBased Enhancing Services for Motivational Support and Behavioral Change. In: *Business & Information Systems Engineering (BISE)* (2013). doi: 10.1007/s12599-013-0273-5.
- [BFRP] Title of the article (2015).
- [CCH*12] Cafazzo, J.A., Casselman, M., Hamming, N., Katzman, D.K., & Palmert, M.R.: Design of an mHealth app for the self-management of adolescent type 1 diabetes: a pilot study. *J. Med. Internet Res.* 14 (2012), 13. <http://dx.doi.org/10.2196/jmir.2058>.
- [CC13] Carmo, M. B., & Cláudio, A. P.: 3D virtual exhibitions. *DESIDOC Journal of Library and Information Technology*, 33 (2013), 222-235.
- [CPG13] Chantzi, A.E., Plessa, C., & Gkanas, I.C.: Design and Development of Educational Platform in Augmented Reality Environment using Gamification to enhance Traditional, Electronic and Lifelong Learning Experience. *Local Proceedings of the Sixth Balkan Conference in Informatics Thessaloniki, Greece (BCI)* (2013), 92.
- [CCH14] Chen, C. Y., Chang, B. R., & Huang, P. S.: Multimedia augmented reality information system for museum guidance. *Personal and Ubiquitous Computing*, 18 (2014), 315-322.
- [CPZS13] Chen, S., Pan, Z., Zhang, M., & Shen, H.: A case study of user immersion-based systematic design for serious heritage games. *Multimedia Tools and Applications*, 62 (2013), 633-658.
- [CKW13] Chi, H., Kang, S., & Wang, X.: Research trends and opportunities of augmented reality applications in architecture, engineering, and construction. *Automation in Construction* 33 (2013), 116-122.
- [Cho14] Choi, H. S.: The conjugation method of augmented reality in museum exhibition. *International Journal of Smart Home*, 8 (2014), 217-228.
- [CFQP14] Clini, P., Frontoni, E., Quattrini, R., & Pierdicca, R.: Augmented reality experience: From high-resolution acquisition to real time augmented contents. *Advances in Multimedia*, (2014). doi:10.1155/2014/597476.
- [DDKN11] Deterding, S., Dixon, D., Khaled, R. & Nacke, L.: From game design elements to gamefulness: defining "gamification". *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. Tampere, Finland: ACM (2011).
- [Fer13] Ferrara, J.: Games for Persuasion: Argumentation, Procedurality, and the Lie of Gamification. *Games and Culture*, 8 (2013), 289-304.
- [Fon11] Fonseca, S.: Embarcações que tiveram berço na laguna – Arquitetura naval lagunar. Porto, Papiro Editora (2011). ISBN 978-989-636-587-5. DL 331260/11.
- [Fon15] Fonseca, S.: Embarcações lagunares Bateiras e Artes (Tombo 2) (2015). Ílhavo, DL 331260/1.
- [Goe13] Goehle, G.: Gamification and Web-based Homework. *PRIMUS*, 23(3) (2013), 234-246.
- [INE14] Instituto Nacional de Estatística. Base de dados estatística de Visitantes (N.º) de museus por Tipologia (2014). Disponível em: www.ine.pt, consultado em 04 de julho de 2015.
- [Jag13] Jagoda, P.: Gamification and Other Forms of Play. *Boundary 2-an International Journal of Literature and Culture*, 40 (2013), 113-144.
- [JQCVB10] Jaramillo, G., Quiroz, J., Cartagena, C., Vives, C., & Branch, J.: Mobile Augmented Reality Applications in Daily Environments. *Revista EIA* (14) (2010), 125.
- [JP12] Jevremovic, V., & Petrovski, S.: MUZZEUM - Augmented Reality and QR codes enabled mobile platform with digital library, used to Guerrilla open the National Museum of Serbia. In: *18th International Conference on Virtual Systems and Multimedia: Virtual Systems in the Information Society, VSMM, Milan* (2012), 561-564.
- [JNRR15] Jiménez, B.F.P., Nex, F., Rizzi, A. & Remondino, F.: ARCube—The Augmented Reality Cube for Archaeology. *Archaeometry*, 57 (2015). 250-262.
- [KPR*13] Keil, J., Pujol, L., Roussou, M., Engelke, T., Schmitt, M., Bockholt, U., & Eleftheratou, S.: A digital look at physical museum exhibits: Designing personalized stories with handheld Augmented Reality in museums. In: *1st International Congress on Digital Heritage, DigitalHeritage, Marseille* (2013), 685-688.
- [KL12] Kim, J. T., & Lee, W. H.: Dynamical Model for Gamification: Optimization of Four Primary Factors of Learning Games for Educational Effectiveness. *Computer Applications for Graphics, Grid Computing, and Industrial Environment*, 351 (2012), 24-32.
- [LCYL06] Lam, A., Chow, K. H., Yau, E. & Lyu, M.: ART: Augmented Reality Table for Interactive Trading Card Game. In *Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications* (2006), 357-360.

- [LGF12] Li, W., Grossman, T., & Fitzmaurice, G.: Gam-iCAD: a gamified tutorial system for first time AutoCAD users. In: Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology. Presented at UIST'12. ACM, Cambridge, MA (2012), 103-112.
- [LGG13] Linaza, M., Gutierrez, A., & García A.: Pervasive Augmented Reality Games to Experience Tourism Destinations. In Information and Communication Technologies in Tourism 2014, Z. Xiang and I. Tussyadiah, Eds. Springer International Publishing (2013), 497-509.
- [Mat12] Matsumoto, T.: Possibility of E-Learning Education That Uses the Gamification. 5th International Conference of Education, Research and Innovation (2012), 3310-3314.
- [MK94] Milgram, P., & Kishino, F.: "A Taxonomy of Mixed Reality Visual Displays", SPIE Vol. 2351, Telemanipulator and Telepresence Technologies (1994).
- [NOCM12] Nee, A., Ong, S., Chryssolouris, G., & Mourtzis, D.: Augmented reality applications in design and manufacturing. CIRP Annals - Manufacturing Technology 61 (2012), 657-679.
- [Nee12] Neeli, B. K.: A Method to Engage Employees using Gamification in BPO Industry. Third International Conference on Services in Emerging Markets (2012), 142-146.
- [RMSN12] Rambli, D., Matcha, W., Sulaiman, S. & Nayan, M.: Design and development of an interactive augmented reality edutainment storybook for pre-school. IERI Procedia 2 (2012), 802-807.
- [RRC*13] Ramirez, M., Ramos, E., Cruz, O., Hernandez, J., Perez-Cordoba, E., & Garcia, M.: Design of interactive museographic exhibits using Augmented reality. 23rd International Conference on Electronics, Communications and Computing, CONIELECOMP, Cholula, Puebla (2013), 1-6.
- [RKW13] Rose, K.J., Konig, M., & Wiesbauer, F.: Evaluating success for behavioral change in diabetes via mHealth and gamification: MySugr's keys to retention and patient engagement. Diabetes Technology & Therapeutics. 15 (2013), A114.
- [SCT*14] Santos, M. E. C., Chen, A., Taketomi, T., Yamamoto, G., Miyazaki, J., & Kato, H.: Augmented reality learning experiences: Survey of prototype design and evaluation. Ieee Transactions on Learning Technologies, 7(1) (2014), 38-56. doi: 10.1109/TLT.2013.37
- [SF14] Seaborn, K., & Fels, D. I.: Gamification in theory and action: A survey. International Journal of Human Computer Studies, 74 (2014), 14-31.
- [SRV13] Simões, J. Redondo, R. D., & Vilas, A. F.: A social gamification framework for a K-6 learning platform. Computers in Human Behavior, 29 (2013), 345-353.
- [TTK*13] Takahashi, T. B., Takahashi, S., Kusunoki, F., Terano, T. & Inagaki, S.: Making a hands-on display with augmented reality work at a science museum. In: 9th International Conference on Signal-Image Technology and Internet-Based Systems, SITIS, Kyoto (2013), 385-390.
- [YW14] Yoon, S. A., & Wang, J.: Making the Invisible Visible in Science Museums Through Augmented Reality Devices. TechTrends, 58(1) (2014), 49-55.
- [ZL10] Zichermann, G. & Linder, J.: Game-based Marketing: Inspire Customer Loyalty through Rewards, Challenges, and Contests. Wiley, Hoboken, NJ. (2010) ISBN: 978-0-470-56223-9.



Posters

Contents

A context-aware immersive interface for teleoperation of mobile robots <i>João Quintas, Luís Almeida, Elísio Sousa, Paulo Menezes</i>	150
3D and the Web - The last twenty years and the future	156
Visualizações Interactivas de Trajectórias em Vídeo no Espaço e no Tempo <i>Ana Jorge, Sérgio Serra, Teresa Chambel</i>	163

A context-aware immersive interface for teleoperation of mobile robots

João Quintas
Instituto Pedro Nunes
Coimbra
jquintas@ipn.pt

Luís Almeida
Instituto Politécnico de Tomar
Tomar
laa@ipt.pt

Elísio Sousa Paulo Menezes
Universidade de Coimbra
Coimbra
pm@deec.uc.pt

Abstract

In this paper we present a context-aware immersive teleoperation interface to assist operators during navigation tasks. This new interface strategy aims to address the problems associated with mental overload, often experienced by operators of teleoperated devices. Our approach simplifies the high complexity of information displayed in control rooms. Our approach includes a context-based human-robot interaction framework that detects relevant information and automatically adapts the displayed interface in virtual windshield. Results showed that the proposed approach enhances user immersion while maximizes task performances and minimizes the operator physical and cognitive workload.

Keywords

Human-Robot Interaction, telepresence, Embodiment, Teleoperation, Context-Awareness

1 Introduction

A telepresence robot [Minsky 80] presents a solution for search and rescue, remote reconnaissance, space exploration or maintenance in contaminated areas. In these scenarios, operators often intervene in the robot control loop when the robot is deployed in remote and unstructured environments [Sheridan 93].



Figure 1. A typical ROV Control Room. Courtesy of Monterey Bay Aquarium Research Institute.

An example of a typical control room for Remote Operated Vehicles (ROV) is depicted in figure 1. In spite of operator's skills and expertise, human decisions in teleop-

eration rely on diverse remote information sources. While teleoperating, operators must be fully focused in their task, which requires processing all inputs and filtering relevant information in order to execute the appropriate action. This intense use of perceptual and cognitive skills may lead to mental and physical strain, which may cause catastrophic hazards. This fact was addressed in [Wickens 08], where the authors studied how humans capabilities vary while performing tasks that require processing information from multiple resources. The studies concluded that multiple sources of information contribute to a high mental workload, causing negative implications on task performance.

In order to reduce the difficulties and stress of teleoperation, several authors propose solutions that allow the user to have a better understanding of the remote environment without the need to keep a mental record of the same.

The sensation of being (inside) the robot improves operator's performance of driving it. Thus, by combining the concepts of telepresence and physical embodiment we are able to create tele-embodiment [Paulos 97]. As result, the operator feels the remote robot body as his own and he/she acts more naturally, minimizing the physical and cognitive workload.

In the other hand, studies showed that operators quite often are not sufficient aware of robot location and surroundings, resulting in most operator decisions are based on remote video information, which forces the operator to try understanding the remote environment through a "key-hole" [Woods 04].

In [García 15], a virtual cockpit was proposed for intervention underwater robots that simplifies the high complexity of information displayed through specifically designed Graphical User Interface (GUI).

To tackle this challenge, we propose an approach to create a context-aware immersive interface for teleoperation of mobile robots that extends typical teleoperation functionalities, allowing human operators to benefit from an improved user experience. Figure 2 illustrates the expected outcome for this approach, which aim to improve over typical control rooms as depicted in figure 1.



Figure 2. Context-aware immersive interface for teleoperation of mobile robots

Consider a teleoperation scenario where an operator is using our immersive interface. The operator is performing a navigation task, of a mobile robot, in remote environment with good weather conditions (e.g. partially sunny and low humidity). For this task, the operator could be more interested in paying attention in bearing and speed of the robot. The immersive interface would display a simplified control panel with widgets relevant only for that given context (i.e. navigating the robot with good weather). Suppose now, that during teleoperation, the weather changed and the robot's environment is rainy with wind gusts. Our context-aware immersive teleoperation interface will now adapt and display widgets related with wind direction and indication of applied torque in the wheels, as a muddy floor requires a more skillful driving to avoid getting stuck.

1.1 Context-awareness

Schmidt in [Schmidt 00] regarded situational context, such as location, surrounding environment or state of the device, as implicit input to the system. This extended the concept of context beyond the informational context into real world environments. The authors used this concept to define the term "implicit human-interaction" as "... an action performed by the user that is not primarily aimed to interact with a computerised system but which such a system understands as input ..."

Röning and Rieki in [Röning 01] proposed a context-aware mobile system, which included mobile personal

robots. They proposed the "Genie of the Net" architecture as an ever expanding system providing helpful information and guidance when human capabilities are exceeded. Their proposed approach also aimed to be a technique to handle several individual robots so that they can co-operate with each other and human beings. Their initial application tests selected the approach of first building a teleoperated robot and then gradually shifting tasks from the human to the robot.

Celikkanat, et.al. in [Celikkanat 15] demonstrated on the iCub platform that using context resulted in an adaptive, online and robust approach for executing two important tasks: object recognition and planning.

In our approach we will define context as:

Context is the set of information that is relevant, affects or constrains how some action is taken, without being the center of interest of the action.

1.2 Contributions and structure

Based on these principles, we present an approach that aims to improve the telepresence experience for the operator when remotely operating a mobile robots. The Augmented Reality based user interface (UI) proposed in [García 15] is now coupled with a context-aware module and will automatically adapt operator's UI to changing conditions that are relevant for the task being performed, resulting in a context-aware immersive interface. This auto-adaptation consists in providing cues to the operator that aim to simplify the teleoperation interface.

The paper proceeds as follows. Section 2 presents design aspects, including teleoperation mechanism and the role of context information in the interaction process of telepresence and teleoperation. Section 3 describes experimental and comparative results of different interface styles. As an application example, we address the scenario of an operator remotely controlling a robot while his context aware user interface adapts to help him during the navigation task, providing the necessary information for the given context while hiding the irrelevant one to avoid distracting or overloading the user. Section 4 summarize the conclusions.

2 Designing the context-aware immersive interface for teleoperation of mobile robots

In this section we will address some design aspects for our approach referring to teleoperation architectural details and necessary adaptations to achieve a context-aware immersive system.

2.1 Teleoperation architecture

Literature proposes teleoperation models with the human operator inside the control loop. Usually, the robot control commands are transmitted through a delayed transmission channel [Islam 14][Sheridan 93][Almeida 14] and, the action feedback is also affected by a transmission delay. The

model purpose is to integrate these delays and keep the robot controllable.

In our research, we explore the relationship between the human and the interface used to control the remote robot. We propose a simplified model composed by an outer teleoperation control loop that uses an inner perception control loop, see figure 3.

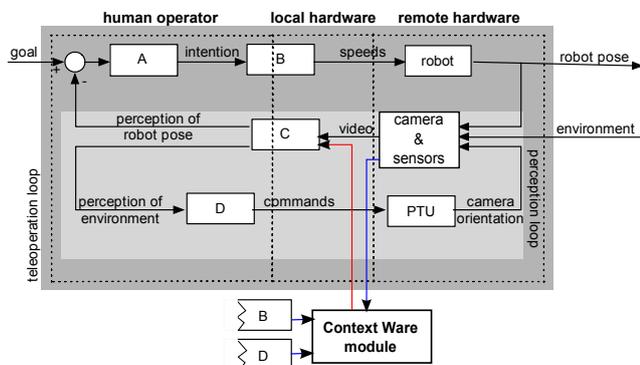


Figure 3. Teleoperation and perception as control loops

teleoperation loop – the robot teleoperation process can be modeled as a standard control loop. Basically, the human operator compares a given goal with the robot’s position in the remote environment. The operator perceives the difference and develops an intention to compensate it, which is later translated into robot’s commands by some interaction system. Figure 3 depicts this loop, where block A represents the perception of the error and the intention generation. This intention is converted into commands through block B, which models the human action into an interface that produces proper robot commands. This control loop will be closed only if the user can perceive the pose of the robot in the remote place.

Perception Loop – this research in teleoperation systems considers a camera point of view, as the operator being inside the robot. The camera’s purpose is to enable user to perceive both the robot motion and the surrounding environment as being driving inside the mobile robot. This perception process can similarly be described as a control loop. In this case, the human operator controls the robot’s camera orientation and utilizes the visual feedback to compensate the scanning process required for a task (ex: track objects, look around, inspect, or navigate). As in the control loop, the camera acquires images and sends them through a channel to the user. This visual information enables the operator to perceive the relative pose of the robot in the remote environment and, the environment itself. Block C represents this process.

The Context Aware module recognizes an activity based on robot’s sensors information, operator’s positional intention of the robot (block B) and operators visual point of view (block D) and, with it selects the useful information to be presented in the operator’s windshield or UI (block C).

2.2 From teleoperation to remote embodied operation

Using the presented model lets map the different perception and control mechanisms into blocks A, B, C and D. We demonstrate how to evolve from a traditional teleoperation concept to new and more immersive approaches. In traditional teleoperation systems, block B represents the robot motion control using a joystick and, block D, represent the control of the pan-and-tilt camera unit using also a joystick. Block C provides the remote images to the user through standard screen, while block A, enables him to convert the positional perceived error into an intention to move the robot.

To create a more immersive interface we propose a viewpoint transfer. To solve the challenge of controlling the remote viewing camera, we suggest the use of a head mounted display (HMD) in which the operator can move his head and almost simultaneously control a pan-and-tilt unit (PTU) that supports the robot’s camera. Block C provides the visual information that enables the user to perceive the difference between the visual goal, and the means to compensate. The human, through block D, acts into camera PTU to gather new point of views.

This type of camera control provides an egocentric view, as the camera movements are synchronous with the operator’s head movements. It enables the user to have an egocentric perception of the remote environment just as if the human was at the robot position and orientation. The described process is a crucial step to give the user the sensation of being physically embodied in the remote robot, which means that “the user will see what the robot can see”.

2.3 Creating a context-aware immersive interface for teleoperating mobile robots

In our approach we consider context recognition to be a periodic process that operates in the background of the system, while interacting with a user. This process is illustrated in figure 4.

It plays the role of detecting changes in the context and control the adaptation in the user interface during teleoperation.

To incorporate context-awareness into our architecture we designed a Context-based Human-Robot Interaction Framework (CB-HRI). Figure 5 illustrates the CB-HRI framework, which conceptually extends teleoperation architecture.

This framework acts as a middleware to integrate contextual information in the overall system and control the workflow related with human-robot interaction.

The main components of the framework are:

1. the *Message Bus* module that includes the interfaces with other components in the architecture.
2. the *Recognition algorithms* module includes the algorithms to match perceived information with contextual information.

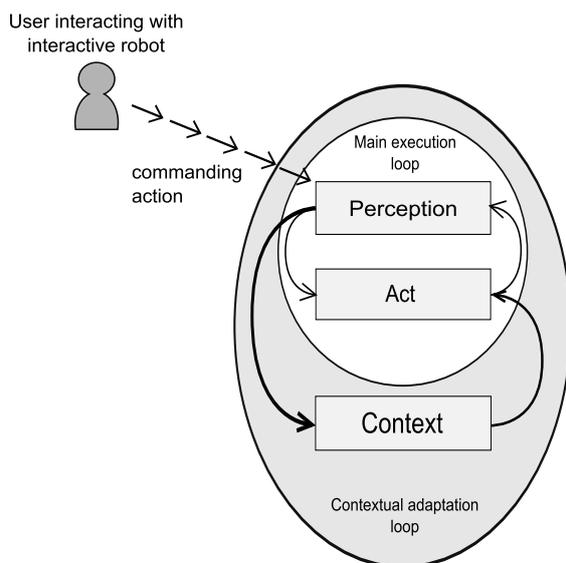


Figure 4. Context verification process

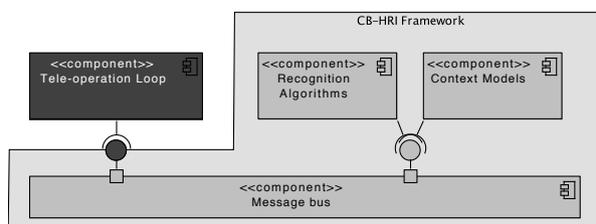


Figure 5. Context-based Human-Robot Interaction Framework extending existing architectures

- the *Context models* is a repository with apriori context data models.

In our approach we propose contextual information as an integration mechanism between a variety of available algorithms and other resources that are known to perform well under a certain conditions.

Therefore, the CB-HRI framework must be integrated with the components dealing with perception, reasoning, data storage and actuation.

3 Implementation and Results

In our experiment the objective was to navigate as quickly as possible, without colliding against walls or obstacles, as illustrated in figure 6. The operator could make the robot move forward, move backward, turn 360° on itself, and control the robot's camera point of view. The robot on-board sensors could provide the following information: movement speed, movement direction, 360° proximity information, camera's pose and battery levels.

Choosing only the most relevant information at any given time, provides uncluttered field of view and decreases the

user's mental workload. Furthermore, for the information that is always present, this can be slightly transparent as not to block the user's view. The graphical elements representing the information should not be too big and placed near the user's view centre, as not to strain the user's eye and focus.



Figure 6. Navigation task, comparison of different teleoperation interaction styles designed to enhance embodiments sensations

3.1 Modeling and recognizing Contexts

In our approach we represent Context as a vector, where each element is a numerical representation of a feature (i.e. context features).

In order to select coherent context features we take as baseline the same measured information in previous works, where we explored immersive teleoperation interaction styles and their application in a virtual cockpit in a navigation task. Thus, we enumerate the context features used as:

- Task (e.g. reserved for future use)
- User proficiency (e.g. 0 = beginner, 1 = amateur, 2 = professional)
- Distance to obstacle (e.g. 0 = close, 1 = near, 2 = far)
- Safe speed limit (e.g. 0 = slow, 1 = fast)
- Bearing to obstacle (e.g. 0 = no adjustment, 1 = adjust right, 2 = adjust left)

In this set of features we can neglect Task, as this information will be irrelevant because we are only performing navigation.

We create a set of rules based on the previous features to define Context classes, as follow:

- Context0: Navigating in open space
- Context1: Narrow space / Close proximity to obstacle

- Context2: Too much speed to avoid obstacle without colliding
- Context3: Wrong bearing to overcome the obstacle

The result of the classification is then used by the user interface, which loads the appropriate widgets that provide relevant information to the user while navigating in a specific environment. Specifically, taking into account our experiment, we have the following adaptations:

- No widgets are displayed in Context0;
- Display the speedometer widget in Context1;
- Display the bearing indicator widget in Context2.

3.2 Implementing the immersive interface

To implement the graphical interface, a combination of OpenGL and OpenCV was used. Setting the video stream as background with OpenCV, the 2D elements are created with OpenGL and then placed on top of the stream. The robot movement speed was designed according to modern speedometers to offer a degree of familiarity to the user, since it is one of the most common and intuitive ways to display an object's speed. As such, a speedometer background containing a gauge was designed and a needle (with a transparent background) was created and placed on top of the background, with the lower end of the needle aligned with its center. By applying rotation to the needle, we can make it move and indicate the robot speed. The robot's direction is simply an arrow indicating forwards or backwards. It is designed to look like it's pointing along the Z-axis (depth) for better intuitiveness (see figure 7). Proximity information is represented as a circle with as many sections as there are sensors. Each section changes color depending on the distance from the robot to the nearest object in the corresponding direction. The circle is designed to look like it's aligned with the Z-axis to facilitate the user's perception of which sensors are displaying information. To create the sections, various circles with single sections are stacked upon each other and controlled independently. The camera pose is represented by a circle inside a square. Pan is represented through the X-axis and tilt is represented through the Y-axis. The square represents the minimum and maximum pan and tilt limits. The battery level is represented by a numerical percentage inside a drawing of a common battery.

Other widgets, like battery level will be displayed if context awareness module considers that is important for the task.

3.3 Validating user interaction

To validate our approach we compared experimentally four interaction styles, which included from traditional joystick approaches to more innovative based on deictic gestures and natural body postures. We carried out a quantitative

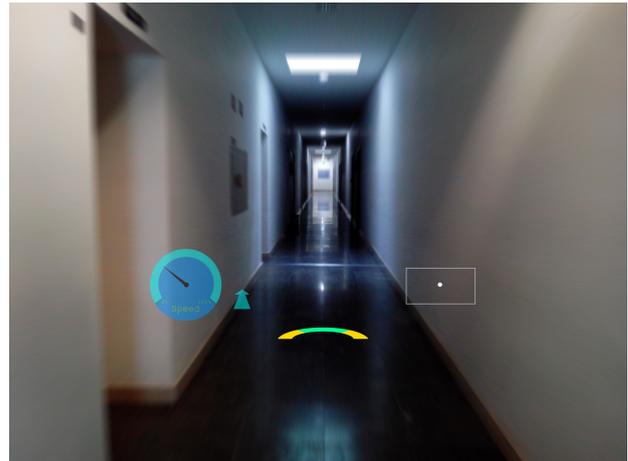


Figure 7. Operator's immersive windshield with smart widgets: semi transparent speedometer, robot direction motion arrow, proximity sensor and camera pose.

and subjective task performance analysis involving 13 participants. All the participants had to teleoperate a mobile robot and navigate through a predefined obstacle course.

The experiment goals were to maximize the task performance and minimize the operator's physical and cognitive workload. We induced in the operator the sensation of being at the remote environment; to generate the remote physical embodiment feeling, the approach consisted in letting user perceive the robot's structure as his/her own body.

To evolve from teleoperation to embodied operation we explored 3 approaches:

- 1) view transfer using an HMD (i.e. with an egocentric controlled view, the user will see what robot can see);
- 2) pointing gestures to control the robot (i.e. user sees him as being the robot, or inside of it, and his pointing gestures are used to control his own motions);
- 3) body posture to control the robot.

To understand the influence of the four different interaction styles on the teleoperation of a mobile robot; and to assess how natural can a user interact and perceive the remote robot structure as his own body, a quantitative and subjective task performance analysis were carry out (13 participants in driving tasks)(figure 6).

Results demonstrated that visual feedback through an HMD improved significantly users task performance (figure 8). The introduction of natural deictic gestures based robot control presented some gain in task performance when compared with joystick. Body intention-based robot control was the operator's choice in all subjective questionnaires, and was confirmed by time performance measures in path driving. As conclusions of the introduced gesture, postures and view control mechanisms improves the physical embodiment sensation. Sensation of controlling the robot from inside reduces mental workload of the opera-

tor. There is a positive effect on user satisfaction and task performance.

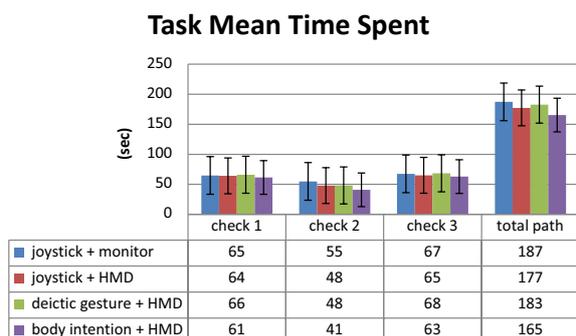


Figure 8. Navigation task performance time comparison while using different teleoperation interaction styles

4 Conclusion and future work

The paper addressed the challenges that enhances telepresence and teleoperation. We considered the importance of integrating contextual information in the interaction process with the objective to improve user experience while performing a teleoperated task.

To understand the influence of 4 different interaction styles on the teleoperation of a mobile robot; and to assess how natural can a user interact and perceive the remote robot structure as his own body, a quantitative and subjective task performance analysis were carry out demonstrating that visual feedback through an HMD improved significantly users task performance.

Moreover, we addressed the navigation task as an example task where the integration of context information can improve usability of the system by providing pro-active cues that help the operator to perform the task. An rational management of the information and a egocentric point of view maximizes task performances and minimizes the operator physical and cognitive workload.

References

- [Almeida 14] L. Almeida, B. Patrao, P. Menezes, and J. Dias. Be the robot: Human embodiment in tele-operation driving tasks. In *RO-MAN, 2012 IEEE*, Aug 2014.
- [Celikkanat 15] H. Celikkanat, G. Orhan, N. Pugeault, F. Guerin, E. Sahin, and S. Kalkan. Learning Context on a Humanoid Robot using Incremental Latent Dirichlet Allocation. *IEEE Transactions on Autonomous Mental Development*, 2015.
- [García 15] Juan Carlos García, Bruno Patrão, Javier Perez, João Seabra, Paulo Menezes, Jorge Dias, and Pedro J. Sanz. Towards an immersive and natural gesture controlled interface for intervention underwater robots. In *Oceans'15 MTS/IEEE Genova, Genova (Italy), 05/2015* 2015.
- [Islam 14] Shafiqul Islam, XiaopingP. Liu, AbdulmotalebEl Saddik, Lakmal Seneviratne, and Jorge Dias. Control schemes for passive teleoperation systems over wide area communication networks with time varying delay. *International Journal of Automation and Computing*, 11(1):100–108, 2014.
- [Minsky 80] Marvin Minsky. Telepresence. *Omni*, pages 45–51, 1980.
- [Paulos 97] Eric Paulos and John Canny. Ubiquitous teleembodiment: Applications and implications. *International Journal of Human-Computer Studies*, 46:861–877, 1997.
- [Röning 01] Juha Röning and Jukka Riekki. Context-Aware Mobile Robots: Part of Smart Environment. In *HRI-workshop at KTH*, 2001.
- [Schmidt 00] Albrecht Schmidt. Implicit human computer interaction through context. *Personal Technologies*, 4(2-3):191–199, June 2000.
- [Sheridan 93] T.B. Sheridan. Space teleoperation through time delay: review and prognosis. *Robotics and Automation, IEEE Transactions on*, 9(5):592–606, Oct 1993.
- [Wickens 08] Christopher D. Wickens. Multiple Resources and Mental Workload. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 50(3):449–455, June 2008.
- [Woods 04] D.D. Woods, J. Tittle, M. Feil, and A. Roesler. Envisioning human-robot coordination in future operations. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(2):210–218, May 2004.

3D and the Web

The last twenty years and the future

Vitor Cardoso
Universidade Aberta (UAb) / CIAC
Lisbon
Vitor.Cardoso@uab.pt

Abstract

A critical view on two decades of 3D graphics “standards” for the Web. Starting with the first W3C standard (VRML) and its evolution (X3D), the reasons for decay and why according to several evidences WebGL, by the Khronos group, seems to be the unavoidable road ahead. Among them, a new phase of “WebGL era” is coming, with full grown 3D applications being WebGL ready.

Keywords

Web3D, Virtual Reality, Virtual Worlds, VRML, X3D, WebGL.

1. INTRODUCTION

In October 2014, HTML’s 5.0 final version arrived. It includes in the norm significant definitions for several media (audio, video, 2D graphics) but, against what was expected by some developers. 3D is absent. At least apparently because, on one hand, VRML/X3D¹ - the “official” norm - was not yet revoked although never widely adopted and still relaying on third party plugins. A X3D Working Group inside W3C and the Web3D consortium have been working for years with “the purpose of fully integrating X3D with HTML” (Web3D Consortium, 2011) but had no success; on the other hand another Web Graphics norm, WebGL, proposed by the Khronos group (well known by its open standard OpenGL) is strongly emerging and already runs plugin less over the major web browsers.

Summarizing, this article addresses the following points:

- VRML/X3D: the first 3D graphics standard for the Web
- VRML/X3D technology: what went wrong?
- WebGL, the road ahead for 3D on the Web?
- Tools for designers with some programming skills
- 3D Graphical editors and Web deployers

2. VRML/X3D: THE FIRST 3D GRAPHICS STANDARD FOR THE WEB

Before any attempt to foresee the future of Web 3D graphics one needs to briefly rationalize and review what has been learned from the history of 3D graphics standards on the Web.

In the first years of the Web era (started in 1991) and the excitement over a multimedia global experience, 3D come naturally as the next level. Proposed in 1994 (Pesce, Kennard, & Parisi, 1994) and endorsed shortly after (Bell, Parisi, & Pesce, 1994) by vrml.org (now web3d.org), an organism under w3c consortium, VRML/X3D technology has been around for two decades as “the” intended 3D graphics standard for the Web. Yet, despite this “official” status, it was never implemented natively on any major web browser. Until today, users must consider installing a plugin in order to run VRML/X3D content inside a browser.

2.1 Enthusiasm in the 1990s

VRML looked, back then, cool and unique for the Web. It supported 3D geometry, animation, and scripting (Web3D Consortium, 2015). From the beginning and especially on the second half of the nineties VRML/X3D gathered considerable attention and enthusiasm among artists, engineers and educators. Since it was doing well and achieving an interesting (moderate) widespread use over the Web, the enthusiasm spread, engaging some reference public and private organizations, from NASA to SUN Corporation. Even today search engines still show a significant evidence of the quantity and quality of VRML work done back then. Beside its own potential and “official” status as a web technology, some key important factors have contributed in our view to this early enthusiasm and use of VRML:

- **General Interest/curiosity for 3D**

In the nineties the curiosity and general interest about 3D

¹ We will use the term VRML/X3D to refer the continuity of this technology with two norm versions that coexist, even today.

was growing and the possibility of associating it to the, also new, Web was appealing.

- **Browser Wars and VRML endorsement**

During the first "browser war" days, in mid-nineties, Netscape and Microsoft had their focus on every relevant novelty to promote their browser and, in no time, Netscape Navigator and Internet explorer had, each one, their free VRML plugin to offer. This was in the public eyes an obvious technology endorsement and clear incentive for users and content developers to install VRML capability on their browsers and use/create 3D for the Web.

- **Multiuser worlds, avatars & 3D Communities**

The third key element was surely the quite immediate availability of 3D multiuser solutions. Innovative 3D multiuser (VRML) worlds with avatars, most using technology from "Black Sun" (latter reborn as Blaxxun), formed at Sun Technologies (Dammer, 1997). Black Sun's worlds open in 1997, Cybertown in 1997 (Poster, 2014) and "Le Deuxième Monde" from 1998 are among the most significant and iconic early projects done in VRML. They gathered on the nineties a significant amount of world users around this technology.

It's important to highlight that in addition to 3D graphics, provided by VRML, Blaxxun and others, brought 3D multiuser technology that constituted a solid base for virtual communities on the Web (Blaxxun Interactive, 1998). The solution was quite *easy* to implement, working nicely and well integrated in the Web ecosystem. This looked back then incredibly innovative and appealing, gathering an enthusiastic community which remained the hallmark of VRML and X3D evolution throughout many development cycles (Butzman & Daly, 2007)

3. VRML/X3D: WHAT WENT WRONG?

In this point we rationalize about how VRML/X3D technology changed from a phase of initial enthusiasm to the present low pace.

In 2001 Web3D.org proposed X3D, a new 3D web graphics norm, meant to solidify the VRML path and ideally be included in Web browsers core, avoiding plugin need, but this never happened. Unfortunately in the early 2000s Netscape had already lost the browsers war and the winner, Microsoft, backed the posture of browser innovation, dropped the support for his own VRML plugin and removed it from their site! Moreover, including VRML in Internet Explorer core was also out of the question.

From then on, there were no more "official" VRML/X3D plugins for specific browsers and users had to take the risk of installing a third party plugin. The opportunity was lost and, despite being an official "3D standard for the Web", since then the interest for VRML/X3D technology dropped significantly.

Cybertown, a vibrant and innovative free community until 2001 was sold to IVN in 2002 and started to charge users

a fee for membership. That led to a massive abandon. At first users and creators looked for other free VRML/Blaxxun related communities and multiuser servers (even Blaxxun had its own free server). ABNet/Babel X3D² was one of them.

In early 2002, Blaxxun went out of business and the support for its free VRML/X3D plugin (Blaxxun contact) was at risk. Bitmanagement took the plugin development, renamed it as bs contact on a new version and ... started to charge for it! The users of their "unrestricted" demo had to cope with an annoying floating logo over the 3D scenes and worlds. Bitmanagement did a fine technical work evolving bs contact to the most recent 3D graphics norms and enhancements, but that ugly logo was probably the last drop that disgusted users and creators.

Many moved away from VRML/X3D to other emergent technology communities, including Second Life³ that was opening its doors in 2002.

From 2002 on, 3D over the web took a low pace but did not die. The "need" was there but now, instead of one, several 3D technologies along with VRML/X3D concurred to fulfill the demand. Among them Unity3D, with its 3D Web plugin and especially Flash.

"2001 saw Adobe's notable rise to web 3D power with version 8.5 of their Director software. Featuring Shockwave 3D technology, Adobe Director allowed creatives to produce hardware-accelerated 3D graphics with scripted interaction using the Lingo language. Full 3D browser based games could be created, such as Xform Games' GoKartGo!Turbo!" (Helix Design Studio, 2013)

For the past decade Flash has been regarded as the de facto standard for deploying rich graphics (including 3D) and multimedia on the web. Unfortunately Adobe professional tools remained out of reach from common 3D non-profit or educational creators since they are expensive and the company never had a policy of free tools (as others like Autodesk do, for example).

3.1 The swan song of VRML/X3D

From the VRML/X3D side, among other interesting projects. Vivaty, a 3D virtual worlds community, deserves special mention. One of the VRML founders, Tony Parisi, took the lead of a brilliant team, including Keith Victor (creator of VRML/X3D editors, Spazz3D/VizX3D/Vivaty Editor) and Rick Kimball (creator of ABNet multiuser server) and founded Vivaty in 2007 (Parisi, 2010). In there we could see a new level of professional quality graphics and innovative social interaction rarely seen before in VRML/X3D. It showed how VRML/X3D was a fantastic Web 3D technology up level with others much more recent. Vivaty closed in 2010 and was, in several ways, the VRML/X3D well deserved swan song.

² <http://www.odisseia.univ-ab.pt/abnet2>

³ Second Life, a non-web 3D technology, uses a specific client program as is out of scope in this paper.

4. WEBGL, THE ROAD AHEAD FOR 3D ON THE WEB?

Now we rationalize about WebGL and the available evidence pointing it as the road ahead.

4.1 What is WebGL?

In mid 2000s the non-profit Khronos group deploys OpenGL ES, a 3D rendering API for mobile and "embedded systems" (ES), based on the desktop long-established 3D rendering standard OpenGL but optimized for mobile/handheld devices. As an industry standard and royalty free, OpenGL ES became universally adopted on small computing devices, most notably phones and tablets to deliver a hardware-accelerated 3D experience.

In early 2009, the non-profit technology consortium Khronos Group started the WebGL Working Group. WebGL is a Web version of OpenGL ES 2.0. The designers felt that, by basing the API on OpenGL ES's small footprint, it would be more achievable to deliver a consistent, cross-platform, cross-browser 3D API for the web (Parisi, 2014).

WebGL is implemented as low-level API JavaScript. It uses the HTML5 canvas element and is accessed using DOM (Document Object Model) interfaces. Automatic memory management is provided as part of the JavaScript language. As such, WebGL runs directly in browsers (desktops or mobiles) without the need for a specific plugin to harness the full power of the computer's 3D rendering hardware. It is today supported by all the major browsers (IE, Firefox, Chrome and Safari) on desktops and mobile platforms..

4.2 Why WebGL? Some relevant aspects

- Technically sound and proved standard. WebGL is based on long experienced, widely adopted, open and free standards and is already supported by all major browsers (IE, Firefox, Chrome and Safari) on desktops and mobile platforms. It is the long waited 3D on the web without plugins!
- It's light and fast. Based on standards with a small footprint, it is more capable to deliver a consistent, cross-platform, cross-browser 3D API for the web and capable to deliver a hardware-accelerated 3D experience using the device GPU directly. We should say here that we were amazed by the incredible speed a demo scene ran on a three year old galaxy note II mobile phone.
- Perfectly integrated in HTML 5 canvas. WebGL wires the GPU to the browser with a JavaScript-based OpenGL ES API, thanks to the HTML5 canvas tag. This means WebGL content is a DOM element (this was never true with VRML, because it operated as a plug-in) and can be manipulated with the same procedural or formatting techniques as any other element. It's finally at reach a seamless 2D/3D web content integration.

4.3 WebGL risks, performance and compatibility

Comparing to VRML/X3D, WebGL as a lower level language is much harder to cope with and direct content creation is for serious 3D graphics programmers.

"WebGL is simply a programming API for JavaScript built on top of OpenGL which is a graphics abstraction layer. Since WebGL makes direct JavaScript calls to OpenGL, content creation is for serious 3D graphics programmers who know how to deal with a 4x4 transformation matrix and who can speak the GLSL shader language fluently. Exposing OpenGL as JavaScript is nice, but we do not expect web-page authors to become graphics programmers ...

X3D describes scene graphs and 3D content declaratively. This means that authors define what geometry and interaction belongs in a model, rather than programming the low-level details for how polygons get built and drawn. Authors can write XML descriptions for their content in a manner similar to (X)HTML.

Therefore X3D authoring is much more like Web development. Content creators can also easily export VRML or X3D models from their favorite authoring tool, from whatever format, and publish them using the Web." (Anita Havele, 2012)

There are some concerns with WebGL security risks coming with direct access to the GPU. Due to these risks, initially Microsoft and Apple refused to support WebGL but that changed in time partially because the browsers war is back again and no one wants to be left behind and also because the upcoming WebGL versions addressed the security concerns. According to WebGL Security white paper, by the Khronos Group, WebGL conforms to all the security principles of the web platform and was designed with security in mind from day one.

WebGL performance is in general not as good as native execution - it is limited by the dynamic nature of JavaScript. Even so, performance has increased over time and the current browser implementations do a great job of optimizing it. In situations mostly GPU-bound, we can now expect WebGL to perform very similar to native code.

Universal WebGL availability and compatibility is growing but, in the meantime, here and there problems may occur. We have experienced problems with Android running in virtual machines, which was to be expected, but problems may also occur, as we also have experienced, on systems with older graphics cards (GPU) and/or outdated drivers. In some cases certain features or all of WebGL isn't available. The Khronos WebGL wiki has a list of supported configurations.

4.4 How are the alternatives to WebGL going?

Comparing to VRML/X3D, WebGL advanced clearly in browser support and it's popular "plugin less 3D graphics" feature is highly appealing. Google Trends shows clearly that in search statistics WebGL term has become much more popular than VRML and X3D immediately after version 1.0 launch in 2011 (Figure 1).

On the other side comparing to Flash it did not. Flash has been an incredible platform, done a lot for the web as an interactivity and entertainment platform. The reasons to change from Flash to WebGL in the foreseen future are more "political" than technical.

The exact timing is not entirely known right now but there

is a declared intention (started by Apple in 2010) to end, sooner or later, the support for plugins in major browsers, especially plugins that run proprietary compiled code. It affects Java, Silverlight, *bs contact* (VRML/X3D), Unity 3D Web plugin, Flash and many others. This is one of the reasons why Unity dropped flash to switch all development efforts in the creation of a WebGL deployer.

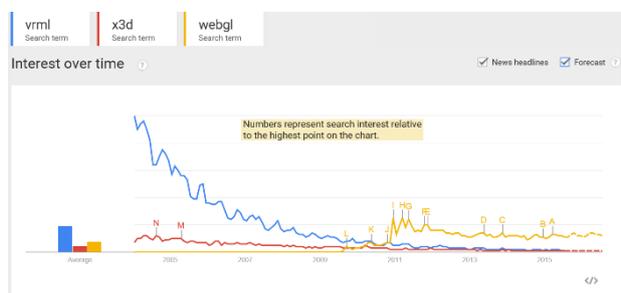


Figure 1 - Comparing VRML, X3D and WebGL Search terms. Source: Google Trends, 2005-2015

Facing an uncertain future, Flash's popularity has been falling in search term statistics (Figure 2).

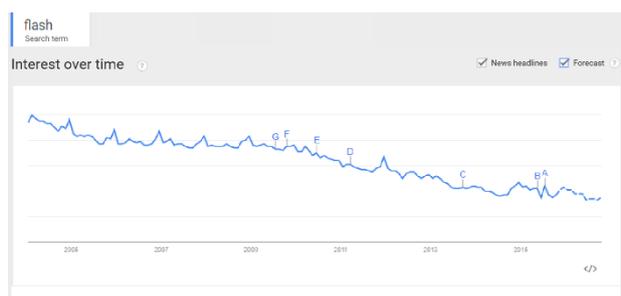


Figure 2 - Flash in Search terms. Source: Google Trends, 2005-2015

4.5 WebGL is already supported on more devices than Flash

"The times they are a-changin'". The declared intentions to end the support for third party plugins in major browsers and the mobile market growth are changing the landscape dramatically and Flash is no longer the largest platform on the planet for interactive browser graphics. Flash is not supported on mobile browsers out of the box; it primarily reaches Mac and PC desktop browsers. On the contrary, WebGL reaches all platforms, desktop and mobile, via the browsers but can also be packaged into native apps and run independent of a browser on both desktop and mobile. It is now supported on more devices than Flash (Krüger, 2014).

With such a wide support for WebGL, the demand for content is growing rapidly and the software industry is already committing. Tool producers are working on WebGL exporters or already have the solutions to offer. Unity 3D stopped Flash export development (the feature was removed in recent versions) and is now committed to WebGL. The recent version 5 outdoes the former Web

plugin by a WebGL export format. Even Adobe itself has adapted its professional tools to produce content in HTML5 and WebGL alongside with Flash. Adobes position is incredibly pragmatic and, obviously, endorses WebGL as the road ahead for 3D graphics on the Web.

In the remaining parts of the article we comment about some fundamental tools for programmers and designers to create WebGL content for the ongoing virtual Web.

5. TOOLS FOR DESIGNERS WITH SOME PROGRAMMING SKILLS

Firstly we comment on tools intended for programmers and designers with some programming skills: the JavaScript middleware libraries for WebGL. Intended for creators with programming skills, they are fundamental to convert and create WebGL content.

WebGL is a low level programming language; this means its code appears very technical and, without a layer of helpful abstraction to assist in reading and learning the syntax, it may frustrate designers with low programming skills who want to produce creative, 3D interactive scenes for webpages. Thankfully a good number of developers have already produced tools and JavaScript libraries to help increase the accessibility of WebGL (a popular one is Three.js). These JavaScript "wrappers" provide an alternative set of commands for creating objects in 3D space. For creators with low programming skills, but that don't fear coding, these libraries simplify the development of WebGL applications and the conversion from other 3D formats to WebGL.

5.1 X3DOM

Web3D Consortium's member, Fraunhofer, using WebGL has developed a JavaScript based interface for X3D intended as a useful framework for WebGL development and transition from X3D. It runs in any HTML 5 browser and supports native X3D within an HTML page. Former VRML/X3D creators will find X3DOM interesting since it converts VRML/X3D objects and scenes to WebGL using a X3D interpreter written in JavaScript. It works well for static scenes and simple animations but, to our knowledge, does not implement the full X3D spec; that would be a major undertaking.

This has been a fine development as the projects available in the site (x3dom.org) demonstrate. Although not exclusively X3DOM is of particular interest to VRML/X3D developers.

5.2 Other JavaScript middleware libraries for WebGL

- Three.js (<http://threejs.org>), is the most popular and has become a reference platform for WebGL development.

A quick browse of the demos page (<http://threejs.org/examples>) shows how powerful and practical this library is. Three.js can convert to WebGL from many popular formats including VRML.

- A list of other JavaScript libraries for WebGL is available at: <http://www.webgl-game-engines.com/>

6. 3D GRAPHICAL EDITORS AND WEB DEPLOYERS

Finally we comment about useful 3D graphical editors to convert and create WebGL content.

6.1 3D editors and WebGL converters

Although JavaScript middleware libraries may be fundamental at some point in the building process, common graphic creators might have a hard time dealing with them. Nothing like using a good graphical 3D editor, indeed. Fortunately a significant number of new editors, tools and plugins (for mainstream 3D editors) appear daily. Look for news and info about them in places such as <http://learningwebgl.com/blog/>.

One of the most interesting new editors is *Coppercube* (<http://www.ambiera.com/coppercube/>), a commercial product. It appeared a couple of years ago and was the first WebGL ready 3D editor. It uses CopperLicht, an open source WebGL library developed by the same owner. Coppercube is not (yet) as full featured as mainstream 3D editors like Blender, but is clean, easy to use and has been until now one of the few tools around that made WebGL content development easy, practical and exciting, as its demos and user forums show.

Unfortunately, the absence of a free editor version for non-profit users and education somehow restrained a potentially larger dissemination that this fine product has merited. Now that the big named editors are committing to be WebGL ready, Coppercube faces a harder fight.

Bitmanagement's BS Content Studio, a commercial product, is another interesting authoring tool that, in the steps of the company's long experience on VRML/X3D, now also exports to WebGL among other format platforms. Unfortunately, also here, the absence of free versions for nonprofit users and education has restrained Bitmanagement fine products from a potentially larger dissemination.

6.2 Free mainstream 3D editors to deploy WebGL

As we see it today, by the products already in place, the ones coming soon and the intentions declared, Unity 3D seems to be best positioned to build and deploy high quality dynamic Web 3D content, whether it's games or interactive virtual scenes and environments, for multiple purposes. Since Unity is not a 3D modeler, designers will need to choose a modeler. Blender and Sketchup are two among several possible options.

6.2.1 3D modellers

Blender is free and one of the best tools for 3D. Does not have (yet) its own native WebGL exporter, out of the box, but we can install the excellent Blend4Web exporter plugin and enjoy its promising WebGL conversions.

Blender is a complete tool and can also create games (includes its own game engine) and interactive virtual scenes. With Blend4Web plugin scenes with limited interactivity can be deployed to the web but not yet games (not yet at least) nor highly interactive virtual scenes. In a WebGL scenario Blender is much more interesting when used with Unity 3D. In this *duo virtuoso* Blender is the modeler for objects and scenes that Unity (which reads blender files

directly) uses as assets to create the action of the game/virtual environment and deploy to end user platforms. Unity manual (Unity Technologies, 2015) has useful info on how to integrate both seamlessly.

One should have in mind that Blender is a serious choice, surely a good one in the long run, but is not necessarily the easiest choice for beginners.

Sketchup instead is a powerful 3D editor, highly intuitive and, from our experience, beginners find it much easier to work with than Blender. Sketchup free version is still a great 3D creation tool and exports to *collada* (.dae), a universal format that Unity3D can import quite well. It has good documentation, instructional videos, a huge collection of free 3D objects and plugins. Adding to this, Sketchup has a legion of followers and tons of free tutorials and resources on the Web.

There are several ways/services to display a Sketchup model in WebGL; the easiest is to upload the model to 3D Warehouse (<https://3dwarehouse.sketchup.com>) which will render exactly this kind of stream automatically and without cost. Once the model has been processed, can be embedded in any web page.

6.2.2 Scene assembler and deployer

The best placed for the moment is probably Unity3D. The reasons why in our vision is a fundamental tool to build and deploy Web 3D content are:

- It is one of the best tools available to deploy games and dynamic virtual environments on several platforms: desktop, mobile and Web. Unity has committed itself to WebGL, starting on version 5 released in early March 2015. Apart from that it deploys content to more than 20 platforms including the former Unity Web plugin, now overtaken by WebGL.
- Until 2014, Unity 3D had a hardly interesting free version since it did not include some essential features of 3D (like dynamic shadows and advanced water, among others) and that disappointed users. That has changed completely and now Unity3D Editor has the same features on free and paid versions. Only some very "high" advanced features, related to special services and game optimization, more directed to professional developers, are reserved for paying customers. In our view, the free version is totally adequate for non-profit users and education.
- In addition, Unity renewed the interface editor (GUI) starting from version 4.6. The 3D editor is now more intuitive and easier to learn.
- Multiuser worlds have for long been created with Unity using its network capabilities, but it was not an easy task so Unity team has committed itself (in the official blog) to deploy with Unity 5.1, recently released, tools and features that simplify the creation of multiuser 3D environments and games, an area much appreciated by former VRML/X3D creators and from other 3D communities.

6.3 A new phase of WebGL era with full grown 3D applications being WebGL ready

We do not want to promote a specific product, but would not be fair to omit that Unity's commitment to WebGL is

not to be seen as one more in a long list. Unity 3D is the first "big named" 3D/game editor to have a full WebGL deployer (not a limited exporter for simple stuff). As a reference tool for games and interactive virtual scenes/worlds creation with almost everything we need, including avatars, special effects, and so on, Unity's WebGL deployer has to be incredibly advanced, a first of a kind, in order to generate correctly all the interaction and special effects demanded by commercial games.

This defines a new era for WebGL where we change from experiments with limited tools (even Coppercube is an ongoing experiment, some relevant features are not entirely implemented or still in development), limited exporters and "specific" tools to full grown 3D applications that offer in WebGL all we had before for other technologies/3D formats. An article about the first commercially available Unity WebGL game (Schwartz & Nyman, 2014) has important information and is a vivid confirmation of what we said about Unity's WebGL deployer relevance.

We have commented here on a selection of tools, most free, to deploy 3D on the Web via WebGL. There are, of course, other alternatives and choices, free or paid, to deploy WebGL content.

7. CONCLUSIONS

VRML appeared in the mid-nineties and was rapidly endorsed by main browser producers through their "official" free plugins. This, joined to a relatively accessible language syntax and the novelty effect of a cool and unique 3D standard for the Web, led to a considerable early enthusiasm that has grown rapidly into powerful multiuser technology developments and the appearance of multiuser worlds and communities of considerable dynamism and influence in the nineties.

The 2000s did not bring the expected boom. Despite the norm upgrade to X3D in 2001, VRML/X3D was never included in major browsers native code. On the contrary, the major browser producer Microsoft, that had recently gained the browser war with Internet Explorer, ended its endorsed free VRML plugin and thus removed millions of users from this 3D technology confining it to a very small market niche. Shortly after, major players like Blaxxun and Cybertown went out of business. Some of their heirs, like Bitmanagement, kept evolving the technology and tried to survive the difficult years but some disregard for nonprofit users, marketing policy and prices, kept moving common users and creators away to other web 3D technologies (among them Flash and Unity3D) and to non-Web as well (like Second Life and Open Sim).

In early 2009, Khronos consortium started the WebGL Working Group and Version 1.0 of WebGL specification was released in March 2011. Developed upon OpenGL ES 2.0 free standard WebGL benefitted immediately from a wide acceptance since it was supported natively by major browsers such as Firefox and Chrome on desktop platforms. The successive announcements of Flash's inglorious deprecation in major platforms also helped WebGL

that widespread rapidly to all major browsers and platforms. The initial lack of good content creating tools for non-programmers has changed over time and now major 3D content creators, like Unity3D, are WebGL ready or aiming to.

Today everything apparently points to WebGL as the major 3D graphics technology running on Web browsers from now on. However, WebGL itself is not a language created for Web designers and direct content creation with the language is reserved for advanced 3D graphics programmers. Middleware JavaScript libraries and mainstream 3D graphical editors reduce this problem significantly but do not eliminate it. At some point, in an exploding 3D on the Web scenario, a Web designer may need to tweak the code directly but WebGL was not designed with that user in mind as VRML/X3D was. Some progress in this regard would be highly welcome

8. REFERENCES

- Anita Havele. (2012, July 6). *Research 2.0 - Thought Leader Interview*. Retrieved from web3d.org: http://www.web3d.org/wiki/index.php/Interview_with_Research_2.0
- Bell, G., Parisi, A., & Pesce, M. (1994). The VRML 1.0 Specification. *Proceedings of the Second International Conference on the World Wide Web, Chicago, October 1994*. Chicago.
- Blaxxun Interactive. (1998). *blaxxun Community Platform - Server SDK*. Retrieved from Blaxxun.com: <http://www.mission-base.com/unweimar/peter/mututorial/api/doc/apextern.html>
- Butzman, D., & Daly, L. (2007). *X3D: Extensible 3D Graphics for Web Authors*. Morgan Kaufman.
- Helix Design Studio. (2013, January 21). *From VRML to WebGL, a look at 3D on the Internet*. Retrieved from blog.helixwebsites.co.uk: <http://blog.helixwebsites.co.uk/2013/01/from-vrml-to-webgl-a-look-at-3d-on-the-internet/>
- Krüger, M. (2014). *Flash Is Dead ... Long Live WebGL*. Retrieved from [wired.com](http://www.wired.com): <http://www.wired.com/2014/05/flash-dead-long-live-webgl/>
- Parisi, T. (2010). *Real-time rants, reflections and ruminations on building a 3D Web*. Retrieved from the *immersive Web(log)*: <http://flux.typepad.com/>
- Parisi, T. (2014). *Programming 3D Applications with HTML5 and WebGL*. O'Reilly.
- Pesce, M., Kennard, P., & Parisi, A. (1994). *Cyberspace. Proceedings of the First International Conference on the World Wide Web, May, 1994*. Geneva.
- Schwartz, A., & Nyman, R. (2014, October 14). *Unity games in WebGL: Owlchemy Labs' conversion of Aaaa! to asm.js*. Retrieved from hacks.mozilla.org:

<https://hacks.mozilla.org/2014/10/unity-games-in-webgl-owlchemy-labs-conversion-of-aaaaa-to-asm-js/>

Unity Technologies. (2015). *Importing Objects From Blender*. Retrieved from Unity Manual: <http://docs.unity3d.com/Manual/HOWTO-ImportObjectBlender.html>

Web3D Consortium. (2011, February 17). *Goals: X3D and HTML5*. Retrieved from Public X3D Wiki:

http://www.web3d.org/wiki/index.php/X3D_and_HTML5

Web3D Consortium. (2015, July). *X3D & VRML, The Most Widely Used 3D Formats*. Retrieved from <http://www.web3d.org>: <http://www.web3d.org/x3d-vrml-most-widely-used-3d-formats>

Visualizações Interactivas de Trajectórias em Vídeo no Espaço e no Tempo

Ana Jorge (*)

Sérgio Serra

Teresa Chambel

LaSIGE, Faculdade de Ciências, Universidade de Lisboa

(*) and Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa
Portugal

{ananunesjorge,sergioserra99}@gmail.com, tc@di.fc.ul.pt

Resumo

Vídeos e filmes estão cada vez mais a ser criados, partilhados e acedidos a partir de diferentes plataformas e dispositivos que permitem a geo-referenciação como forma de enriquecer a sua contextualização, requerendo novas possibilidades de procura, navegação e acesso. O desafio é tornar esta procura eficiente, dada a enorme quantidade de itens disponíveis e a sua inerente complexidade. A visualização interactiva, em complemento à sua catalogação, pode ajudar a responder a este desafio, tornando a informação mais acessível e útil. Acreditamos também que a eficiência da visualização aumenta se, ao mesmo tempo que proporciona novas maneiras de procurar e filtrar através das propriedades dos vídeos, o processo for lúdico e esteticamente interessante.

Em trabalho anterior focámos a dimensão temporal em vídeos e filmes. Tornámos possível explorar e aceder aos filmes e vídeos por critérios de tempo, género e classificação pelos utilizadores, e aceder aos conteúdos relacionando imagem, movimento, áudio, palavras proferidas e representadas em legendas e emoções. No presente trabalho ampliamos o nosso objectivo e consideramos a dimensão espacial através de trajectórias tais como as que encontramos nas visitas guiadas em cidades. Apresentamos o design das principais visualizações interactivas que consideramos chave para navegar em vídeos geo-referenciados, e que permitem: 1) uma visão geral dos vídeos filmados numa dada localização geográfica num dado momento no tempo; 2) aproximar a visão geral para ver os vídeos individualizado, por critérios tais como a distância percorrida, a velocidade, a idade e o conteúdo da filmagem; e 3) acesso ao conteúdo do vídeo da trajectoria escolhida por critérios de cor, áudio, palavras proferidas, emoções e outras trajectórias vizinhas filmadas na mesma zona.

Palavras-chave

Visualização Interactiva, Acesso a Filmes e Vídeos, Visitas Turísticas e Trajectórias, Mapas, Espaço e Tempo.

1. INTRODUÇÃO

Os avanços na tecnologia de vídeo estão a permitir gerar, partilhar e aceder a vídeos e filmes em grandes quantidades, o que exige encontrar novas formas de os procurar, navegar e aceder. Estes vídeos são, cada vez mais, acedidos a partir de diferentes plataformas e dispositivos, e cada vez mais o vídeo pode ser geo-referenciado, permitindo enriquecer a sua contextualização.

A riqueza e complexidade que fazem do vídeo um meio tão interessante tem inerente uma complexidade desafiante com que temos de lidar - o facto de não ser estruturado e correr, ou variar, ao longo do tempo, tornando difícil a tarefa de perceber e procurar todo o conteúdo do vídeo. Este facto torna-se ainda mais complexo na presença da grande quantidade de vídeos que é disponibilizada ao longo do tempo. A visualização interactiva pode ajudar a resolver este desafio de forma eficaz, estética e lúdica.

O foco deste trabalho são as trajectórias na sua dimensão Espaciotemporal. [Peuquet94] definiu ‘trajectória’ como

um evento espacial constituído por sequências de eventos espaciais que cruzam o espaço e o tempo. As visualizações que agora apresentamos, pretendem adicionar o espaço à dimensão temporal anteriormente explorada em visualizações interactivas que permitem explorar e aceder a vídeos e filmes, por critérios de tempo de saída para o mercado [Jorge12] e de tempo dentro do conteúdo dos vídeos onde as suas propriedades audiovisuais são representadas de forma sincronizada [Jorge13]. A nossa motivação tem duas vertentes:

1. Extensão do MovieClouds [Gil12], uma aplicação interactiva baseada na web que adopta o paradigma de *tag clouds*, e que permite explorar e aceder a filmes através da informação disponibilizada, acima de tudo pelo áudio e legendas onde a semântica é, na sua maior parte, expressa, e para o qual já explorámos e enriquecemos o suporte para as dimensões visual e temporal [Jorge12,13];
2. Extensão do SightSurfers [Noronha12] [Ramalho13], uma aplicação interativa baseada na web para parti-

lhar, visualizar e navegar vídeos interactivos geo-referenciados, filmados a 360º pelos utilizadores e incluindo visitas em cidades ou actividades mais radicais como corridas de *kart*. Estes vídeos podem ser experienciados em maior imersão e isolamento, ou sincronizados com um mapa enquanto decorrem.

Cruzando estes trabalhos, focamos a dimensão espaciotemporal dos vídeos e a riqueza dos seus conteúdos, quer gerados pelos utilizadores, quer em filmes, com um foco especial nas trajectórias tais como as que encontramos nos guias turísticos.

Os cenários de uso para visualizações focadas em trajectórias são diversos: entretenimento, aprendizagem e o campo das artes, quer em contextos profissionais ou mais amadores ou casuais. Acreditamos que pode ser interessante, útil e até divertido saber informações sobre eventos e trajectórias, tanto num contexto profissional e preciso (e.g. a zona mais filmada em Lisboa durante a noite ou a relação entre a zona mais filmada e a mais barulhenta e colorida); ou uma procura mais informal (e.g. um turista interessado no local mais calmo e verde da cidade).

Apresentamos estudos de design de visualizações interactivas que permitem navegar e aceder a informação agregada sobre trajectórias em espaços de vídeo (coleções) e vídeos individuais segundo critérios tais como: quantidades de vídeos filmados numa determinada zona, a velocidade da trajectória filmada, a idade da filmagem e o seu conteúdo em termos de cor, áudio, palavras preferidas incluídas nas legendas, e quantidade de movimento, explorando os conceitos de representação de tempo e espaço, essencialmente em mapas.

A secção 2 apresenta o enquadramento conceptual, seguida do trabalho relacionado mais relevante em visualização espaciotemporal e geo-referenciada na secção 3. Na secção 4 descrevem-se e exemplificam-se as nossas propostas de visualização espaciotemporal de trajectórias em vídeo, e o artigo termina com uma reflexão em forma de conclusões e considerações sobre direcções futuras.

2. ENQUADRAMENTO CONCEPTUAL

Movimento é a alteração no tempo de posições espaciais de um ou mais objectos (*movers*) [Andrienko11] [Dodge08] [Parent13]. O foco do trabalho que agora apresentamos são as trajectórias geo-referenciadas em vídeos. É sustentado na taxonomia que relaciona os tipos de dados espaciotemporais com o tipo de análise a efectuar [Andrienko11] e com um enquadramento conceptual [Peuquet94,02] que separa os componentes dos dados espaciotemporais, da seguinte forma: 1) espaço (onde), 2) tempo (quando), e 3) objectos (o quê), conceito destrinchado pelas seguintes questões: 1+2) 'quando' + 'onde' leva-nos a 'o quê' e descreve os objectos numa determinada localização num dado espaço de tempo; 2+3) 'quando' + 'o quê' leva-nos a 'onde' e descreve locais ocupados por determinados objectos em dados momentos no tempo; 1+3) 'onde' + 'o quê' que resulta em 'onde' e descreve o tempo que os objectos ocupam em determinado local.

Como extensão de trabalho anterior no qual foram usadas representações abstractas na dimensão temporal

[Jorge12,13], exploramos agora o espaço e a representação por mapas. Este tipo de representação é visto como a solução chave na visualização de objectos ou fenómenos que consideram a dimensão espacial [Peuquet01] [Kraak10] por não apenas os situarem em determinada localização geográfica mas também os relacionarem entre si [Peuquet01]. Acreditamos que os mapas podem estabelecer uma associação directa e simples entre a localização de interesse para o utilizador, capturada nos vídeos, e a sua representação visual, ajudando a relacionar de forma eficaz a localização dos espaços real e representado.

As visualizações interactivas de vídeo trazem desafios complexos, e em especial quando se lida com o espaço e o tempo em sincronia. A nossa intenção é tornar possível a navegação pelos vídeos e filmes, tanto no que respeita a colecções como a itens, permitindo a associação do seu conteúdo à sua localização particular, ao longo do tempo, e por conseguinte, uma filtragem da informação que seja capaz de trazer resultados mais ricos e precisos.

3. TRABALHO RELACIONADO

O trabalho que mais se assemelha ao nosso segue o conceito espaciotemporal e pode ser baseado, ou não, na representação em mapa.

3.1 Visualização Espaciotemporal

As representações estáticas que se seguem permitem representar o espaço de um modo abstracto e descrevem os conteúdos de vídeo ao longo do tempo através da técnica de sumarização.

Flickr Flow [Viégas09] representa as cores dominantes em fotografias tiradas ao longo de um ano. O resultado visual é a sùmula dessas imagens dispostas circularmente no sentido dos ponteiros do relógio, com as diferentes estações representadas pelas respectivas cores. A técnica *slit-scan* [Levin05], frequentemente usada em arte digital, é utilizada no *What Did I Miss?* [Nunes07], um sistema de visualização baseado numa linha de tempo (*timeline*) que permite explorar visualmente a história de participação captada em vídeo, de modo a dar a percepção sobre a presença e a actividade das pessoas num contexto colaborativo. A ideia é encorajar a conexão entre os participantes enquanto trabalham. *Last Clock* [Angesleva05] também permite guardar o relato visual dos acontecimentos num determinado local, através do ritmo, som e movimento no momento em que aconteceu. *Artifacts of the Presence Era* [Viégas04] mostra os eventos num museu ao longo do tempo através de imagens captadas em vídeo, imagens que veêm alterada a sua altura dependendo da amplitude do som ambiente.

3.2 Visualização de *media* geo-referenciada

Os trabalhos que se mostram em seguida focam informação geo-referenciada. Em alguns casos é possível aceder ou navegar pelas colecções e conteúdos de vídeos e filmes. O Trabalho de [Cruz11] apresenta uma visualização dinâmica da evolução dos dados geo-referenciados retirados da velocidade do trânsito nas ruas, sendo que a largura, a cor e o comprimento das linhas que representam os percursos são alterados pelo número de veículos e pela sua velocidade média. Com foco na orientação e

baseado na representação por mapa, o Google Street View permite ver imagens fotografadas a 360° através de uma projecção esférica com geo-localização. Panoramio (.com) permite seleccionar fotografias de locais num mapa e filtrá-las por etiquetas (*tags*). Nenhuma destas aplicações, no entanto, contempla o vídeo. O espaço e o tempo são também abordados por [Andrienko11] embora não especificamente para vídeo.

A aplicação Movie Mashup MoMa [Finsterwald12] mistura informação do DBpedia e GeoNames, permitindo encontrar filmes através da sua localização mas não contempla trajectórias. [Hao11] relaciona vídeos gerados pelo utilizador com a sua área geográfica, num mapa. Selecciona automaticamente *keyframes* por forma a representar vídeos baseado na popularidade da localização. A ideia é salientar pontos geográficos de interesse, exclui no entanto as trajectórias.

No SightSurfers [Noronha12] [Ramalho13], concebido e desenvolvido no nosso projecto, os utilizadores podem captar, publicar, partilhar e procurar vídeos filmados a 360°, ao longo de trajectórias que podem ser vistas e navegadas num mapa e sincronizadas com o vídeo que está a ser visionado, em qualquer momento. Uma marca visual mostra a presente localização e o utilizador pode seleccionar uma zona da trajectória para ver o vídeo nesse ponto geográfico e nesse tempo específico. O vídeo apresenta linhas coloridas num mapa do Google, mas sem representação visual de velocidade, idade e conteúdo dos vídeos, dimensões que desenvolvemos agora.

4. VISUALIZAÇÃO ESPACIOTEMPORAL DE TRAJECTÓRIAS EM VÍDEO

Apresentamos nesta secção o *design rationale* e os estudos de design das visualizações interactivas de trajectórias de vídeo. O trabalho foca as dimensões espaciotemporais e foi concebido para navegar vídeos geo-referenciados de trajectórias, tanto no espaço de vídeo no que diz respeito a informação aglomerada, ou agregada, como aos próprios conteúdos dos vídeos, que podem também ser acedidos individualmente e visionados.

4.1 Design Rationale

Os filmes e os vídeos têm um grande poder de nos afectar perceptual, emocional e cognitivamente pela história que contam e em grande parte pelas cores, som e movimento que apresentam. Algumas trajectórias podem não ser explícitas no seu papel de contadoras de histórias mas mesmo assim incluem experiências que as pessoas gostam de partilhar e aceder ao longo do tempo. Por outro lado, os mapas ajudam na compreensão da dimensão espacial, e como modelos da realidade, quando são ligados ao vídeo oferecem ao utilizador uma nova visão do real [Peuquet01] aumentando o poder dos vídeos em afectar a percepção humana. Acreditamos que a contemplação das dimensões espacial e temporal na visualização de vídeo enriquece e melhora a percepção dos utilizadores quando acedem e visionam os vídeos.

Nas visualizações que propomos, focamos as trajectórias de vídeo num dado espaço geográfico, representadas em

mapas, e os seus atributos multidimensionais que variam ao longo do tempo.

Acreditamos na utilidade da visualização com dimensões espaciais e temporais no sentido de enriquecerem a visão geral (*overview*) dos vídeos, bem como a sua navegação e acesso, e em especial os que envolvem percursos ou trajectórias. De salientar, a possibilidade de poder assim conhecer, de forma contextualizada, experiências captadas e partilhadas pelos utilizadores ao longo destas localizações. A nossa intenção é explorar o poder da visualização em caracterizar locais visualmente, gerando impressões, emoções e um pensamento no observador, para além de se permitirem análises mais exactas através de informação mais precisa. Informação como o número de trajectórias filmadas pode ser necessária, por exemplo, para analisar as preferências dos turistas no que diz respeito aos locais que eles consideram de maior interesse na cidade, ou se preferem visitar a cidade de noite ou de dia, de forma mais lenta detalhando a informação, ou mais rápida, para uma ideia mais geral. Podem ainda querer saber se a rua em que estão é a mais popular da cidade, ou se é mais calma ou com um ambiente mais verde que as ruas vizinhas. Um estudante, ou um professor podem precisar de informação sobre uma dada cidade, e em vez de ver todos os filmes ou vídeos com o nome dessa cidade no título ou através dessa palavra-chave, podem filtrar a informação por localizações mais precisas e pelas propriedades dos conteúdos dos vídeos.

Conceptualmente, este trabalho baseia-se na metáfora das pegadas que as pessoas deixam no chão. Como se, enquanto filma um trajecto, o utilizador deixasse visíveis as emoções que sentiu através do que viu e ouviu ao longo do trajecto.

Apresentamos estudos de design de visualizações interactivas, concebidas para navegar vídeos geo-referenciados ao longo de trajectórias, através de propriedades qualitativas e quantitativas dos seus conteúdos, da visão geral (*overview*) aos vídeos individualizados através de: 1) visão geral das colecções de vídeos, com informação potencialmente agregada; 2) visão geral dos vídeos individuais na colecção de vídeos; e 3) possibilidade de aceder ao conteúdo de cada vídeo de forma global ou agregada, ou mais detalhada, ao longo do tempo, e visioná-lo, como se descreve em seguida.

4.2 Visualizar Quantidades de Vídeos

A visualização da Fig. 1 mostra os vídeos filmados numa determinada área geográfica. Esta visão geral dá ao utilizador uma ideia sobre a quantidade de vídeos e filmes realizados num determinado limite de espaço e de tempo (de acordo com os critérios de pesquisa) num enquadramento no mapa da Europa. Nesta visão geral observamos que França e Inglaterra são os países da Europa com maior número de filmagens nas condições pesquisadas (sendo a quantidade reflectida na dimensão dos pontos que as representam).

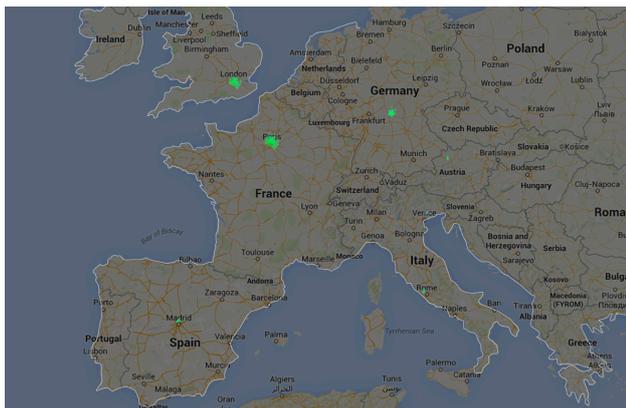


Figura 1: Visão geral das trajectórias filmadas numa dada região geográfica na Europa.

A vista da Fig.1 pode ser aumentada (através de selecção ou zoom), e nesse caso a visualização mais detalhada será constituída pela disposição das ruas em vez dos pontos e nomes das cidades (Fig.2).

A visualização da Fig.2 é uma vista aproximada de uma zona da cidade. Representa um aglomerado de percursos e permite a identificação individual das trajectórias no seu todo. À cor mais brilhante correspondem mais filmagens. Os utilizadores podem filtrar a procurar por critérios relacionados com o espaço e o tempo (e.g. a zona de Lisboa mais filmada às 21 horas). É possível saber que parte da cidade foi mais filmada em geral, ou precisar a informação por hora específica do dia ou da noite, em ambiente de sol ou chuva, pela idade da filmagem, a velocidade da trajectória, a trajectória preferida pelos utilizadores (com maior classificação) ou a que foi mais procurada e vista. O critério pode ser quantitativo (e.g. a maior quantidade de vídeos filmados) ou categórico (e.g. de dia ou na Primavera).

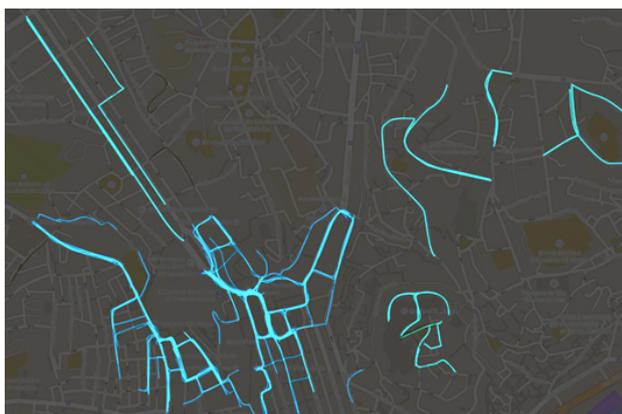


Figura 2: Visão geral das trajectórias filmadas numa dada localização geográfica num dado momento no tempo.

As visualizações das Fig.1-2 representam o espaço no mapa. O tempo é inerente às trajectórias não estando explícito, no entanto, nestas visualizações. O tempo pode ser o critério de escolha do vídeo (e.g. filmado em determinado mês). Como referido, para diferentes níveis de detalhe da informação, é possível aproximar ou afastar

a vista (Fig.1-2), e pode arrastar-se o mapa com o dedo, para visualizar outras localizações. Também é possível optar por mudar a vista, ou perspectiva de visualização (exo, Fig.2-3).

4.3 Visualizar Velocidade em Trajectórias

Foi concebida uma representação 3D para representar a velocidade das trajectórias e a idade das filmagens numa dada localização geográfica (Fig.3). O utilizador pode seleccionar um percurso, individualizando deste modo a informação sobre essa trajectória, permitindo saber as suas propriedades (Fig.4).

A velocidade da trajectória é representada pelas diferentes alturas dos arcos, correspondendo a mais alta ao trajecto que demorou mais tempo a percorrer, como se fosse constituída por mais *frames* e por essa razão contivesse maior quantidade de filme, ou seja, de linha. O tom de verde indica a idade da filmagem sendo a mais brilhante, que tende para o branco, a mais recente e a mais escura, a mais antiga.

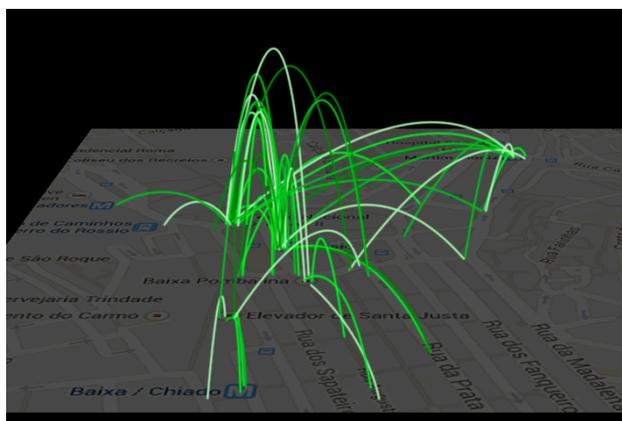


Figura 3: Visualização 3D que representa a velocidade e a idade da filmagem das trajectórias. Uma linha mais alta/menor corresponde a uma trajectória mais lenta, e um verde mais claro/vivo corresponde a uma trajectória mais nova/recente.

Arrastando o dedo sobre a visualização, o utilizador pode alterar a perspectiva da vista podendo comparar com maior precisão os arcos mais altos e mais baixos, que correspondem aos percursos mais lentos e mais rápidos, respectivamente. Nesta visualização enfatizam-se o início e o final da trajectória, ficando em segundo plano as ruas específicas no mapa visto que esta informação está presente na visão geral (Fig.2) e pode também ser vista em detalhe na vista individualizada (Fig.4), seleccionada por duplo clique sobre a trajectória na Fig.3.

Cada trajectória tem as suas coordenadas no mapa e a informação mais precisa sobre quantidades pode ser ligada ou desligada pelo utilizador, dependendo do seu interesse numa informação mais ou menos precisa. O utilizador pode ainda preferir esconder a informação de modo a ter uma ideia mais clara das relações entre o comprimento e a idade dos vídeos.

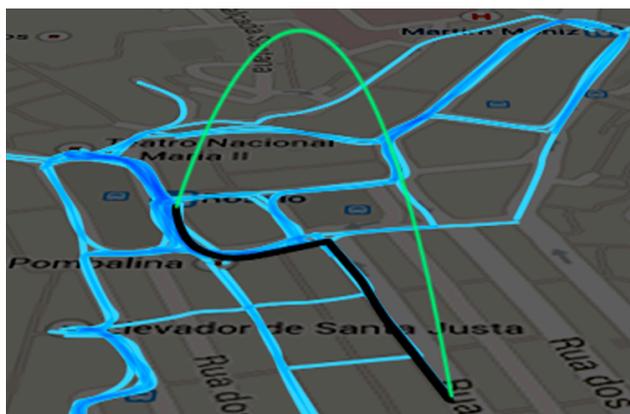


Figura 4: Visualização 3D de um percurso individual. Permite uma percepção mais clara e precisa da rua da trajetória (linha preta), a sua duração (altura da linha) e a idade da filmagem (quantidade de brilho no tom de verde).

Clicar na trajetória de interesse permite uma informação mais detalhada sobre essa trajetória (Fig.4). O duplo-clique torna possível navegar para a visualização que mostra informação sobre o conteúdo do vídeo (Fig.5).

4.4 Visualizar Conteúdos de Trajetórias

Esta visualização (Fig.5) apresenta o conteúdo de uma trajetória através das seguintes propriedades:

1. A posição geo-referenciada no mapa permite ao utilizador saber a localização precisa, o nome da rua, o início e fim do percurso;
2. As cores da trajetória filmada estão presentes por sumarização do vídeo permitindo *e.g.* saber se o percurso é escuro ou claro, possivelmente se em exterior ou interior, se em ambiente de natureza ou cidade;
3. A amplitude do som do vídeo está representada pelo contraste de cor azul sobre a sumarização da trajetória. Não está muito evidenciada devido à prioridade conferida às cores da trajetória mas pode ser tornada mais ou menos visível pelo utilizador aumentando ou diminuindo a opacidade, de modo a obter a transparência (quando a quantidade de som não é importante) ou totalmente opaco (quando a quantidade de som é mais importante que as cores da trajetória). Esta informação dá uma ideia do nível de barulho ou silêncio da trajetória;
4. As palavras proferidas pelas pessoas que gravaram o vídeo tanto quanto pelas pessoas que com elas se cruzam, são gravadas ao longo do vídeo e representadas através de *tag clouds* que representam as frequências. São representadas a branco ao lado da trajetória e podem contribuir para caracterizar o percurso caso se relacionem com o trajecto filmado, e sempre permitirão saber sobre que falam as pessoas quando gravam percursos naquela localização;

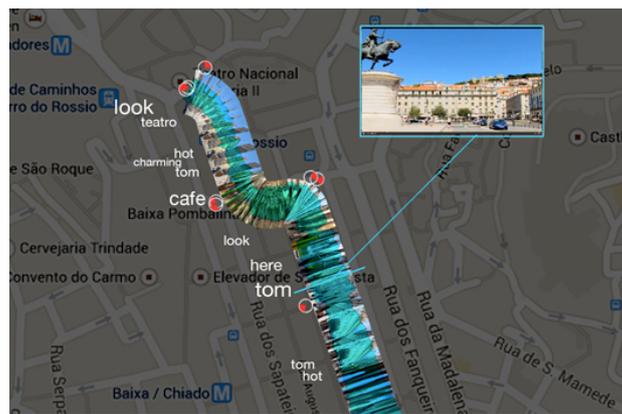


Figura 5: Visualização dos conteúdos ao longo de uma trajetória.

5. As conexões vizinhas a partir do trajecto escolhido mostram outras possibilidades de percursos que o utilizador pode seguir a partir do ponto geográfico em que se encontra. São representadas por pequenos círculos brancos com vermelho adicionado que indicam o início e a direcção do percurso filmado e que o utilizador pode escolher para seguir;
6. Um qualquer momento na sequência do vídeo pode ser seleccionado pelo utilizador de modo a obter uma ideia real da trajetória. Mantendo o cursor sobre a trajetória podem ver-se as *frames* de interesse nesse ponto preciso e com um duplo-clique, o vídeo corre.

Esta vista pode ser aproximada para maior detalhe (aproximar a sumarização da imagem da trajetória torna mais visível o conteúdo das suas *frames*).

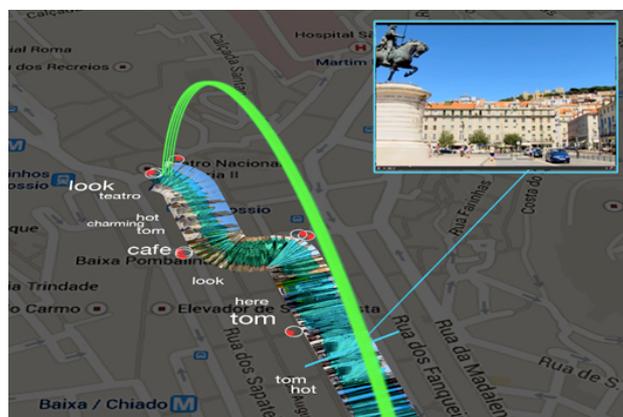


Figura 6: Visualização em 3D de um percurso individual com informação sobre o conteúdo, a velocidade e a idade da trajetória.

Todas as propriedades representadas podem ser ligadas ou desligadas dependendo da intenção do utilizador, e como referido anteriormente, a transparência que representa o áudio pode ser ajustada sendo possível enfatizar essa informação através da quantidade de opacidade da cor azul. A transparência tem, assim, dois objetivos: funcional e estético.

É também possível alterar a perspectiva da visualização arrastando o cursor, modificando a dimensão de 2D para 3D, possibilitando deste modo adicionar informação sobre a velocidade e a idade da filmagem da trajetória (Fig.6).

4.5 Propriedades dos Conteúdos dos Vídeos

A visualização na Fig.7 permite ter uma visão geral do conteúdo do vídeo, procurar informação particular relativa às suas propriedades e visionar o vídeo.

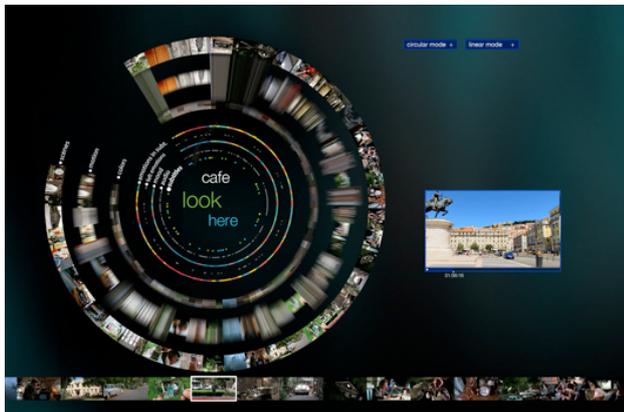


Figura 7: Visualizar conteúdos de um filme ou vídeo.

Os elementos circulares encontram-se em sincronia com a *timeline* abaixo que representa a sequência do filme em imagens com maior detalhe. Enquanto os elementos circulares constituem o filme na totalidade podendo, deste modo, ter-se a ideia geral do filme, a *timeline* representa a cena seleccionada e as vizinhas com maior clareza.

Rodando os elementos circulares eles movem-se em sincronia com a *timeline* alinhando verticalmente a cena e as suas propriedades.

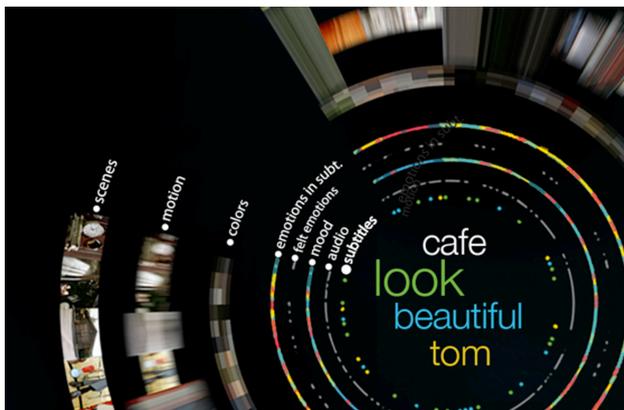


Figura 8: Visão aproximada da visualização da Fig.7.

Apresentamos uma vista aproximada da visualização na Fig.8, e explicamos de seguida as propriedades.

1. Palavras proferidas ao longo da filmagem;
2. Eventos sonoros;
3. Estado de espírito do filme obtido através do som;

4. Emoções sentidas pelas pessoas que viram o filme, medidas biometricamente;
5. Emoções expressas nas palavras proferidas;
6. Cores dominantes;
7. Quantidade de movimento;
8. Imagem da primeira *frame* de cada cena.

É possível ver várias propriedades em simultâneo, para melhor perceber o que acontece em simultâneo (exo. o estado de espírito, ou a cor dominante, quando é proferida uma certa palavra), ou escolher apenas as propriedades de interesse para o utilizador escondendo as restantes, de modo a tornar mais evidente a compreensão da informação. Pode ainda visionar-se o filme em sincronia com as propriedades representadas nos elementos circulares da visualização, e a par com a *timeline* abaixo.

No centro encontra-se uma tag cloud que indica a frequência de forma agregada para o filme inteiro da pista de informação seleccionada pelo utilizador (no exemplo são as palavras proferidas). Para todas as pistas, é representada informação ao longo do tempo, correspondendo aos momentos em que ocorrem. Como no trabalho prévio MovieClouds, as propriedades na pista seleccionada pode estar sincronizada com a *tag cloud* central: ao longo da apresentação do vídeo, cada ocorrência no vídeo causará o *highlight* ao centro da *tag* correspondente (exo. 'café' nas palavras proferidas ou faladas) cada vez que ocorrem; em sentido inverso, quando é seleccionada uma *tag*, são evidenciadas na mesma cor (através de pontos) as ocorrências ao longo da *timeline* circular correspondente.

A partir desta visão geral (Fig.7 e 8) é possível ter uma ideia do vídeo e das suas propriedades. Para mais detalhes sobre esta visualização, veja [Jorge13].

Na visualização da Fig.9 podem ver-se as propriedades de uma dada cena de um filme que tem uma trajetória filmada. Particularizando o caso de um utilizador que pretende visionar a trajetória de um filme ou vídeo, é possível escolher a cena que inclui essa trajetória e visioná-la. Podem ver-se as propriedades nos elementos circulares sincronizados com a *timeline* abaixo e, a decorrerem, a cena do filme (à direita e cima) e a trajetória georeferenciada no mapa (à direita por baixo).

Vistas como um todo, e com um foco particular nas trajetórias, as visualizações complementam-se entre si, permitindo o seu cruzamento conforme a intenção do utilizador relativamente à informação de interesse. Mais, permite inverter o sentido da navegação do geral para o particular (e.g. da informação sobre quantidade de trajetórias filmadas na Europa (Fig.1) para a escolha da trajetória individual (Fig.6-9) assim como do particular para o geral (e.g. a partir de uma cena de um filme saber quantas trajetórias foram filmadas nesse mesmo espaço geográfico).



Figura 9: Visualização de conteúdos de uma cena do filme ou vídeo (a ser visionada à direita em cima), a par com a trajectória percorrida (à direita em baixo).

4.6 Exemplo de Interação

Um visitante de Lisboa pretende ter uma ideia dos lugares mais filmados da cidade, pois isso pode corresponder aos locais mais interessantes a visitar (Fig.2). Pode querer saber quais, dos trajectos filmados, são os mais longos em tempo, no sentido em que estes serão provavelmente os que têm maior detalhe e os mais recentes dada a sua intenção de ter uma ideia aproximada do lugar, nesse momento (Fig.3). Querendo aprofundar a informação, o visitante escolhe um percurso e pode ver a sua ambiência relativamente a cores, som, palavras proferidas ao longo do vídeo, assim como possibilidades de outros percursos de interesse que poderá seguir a partir desse (Fig.5). Pode ter curiosidade em saber que emoções outras pessoas sentiram quando viram esse vídeo (Fig.7). Pode ainda visionar o vídeo para efectivamente saber que lugar vai encontrar ao fazer esse percurso, como se estivesse lá.

5. CONCLUSÕES E TRABALHO FUTURO

Apresentamos neste artigo o nosso trabalho mais recente que versa a visualização interactiva de trajectórias geo-referenciadas. Em trabalho anterior explorámos a componente temporal combinando informação sobre filmes num espaço cronológico (quando saem para o mercado) e nas propriedades dos conteúdos. Neste momento, consideramos trajectórias geo-localizadas e apresentamos estudos de design de visualizações para navegação geo-referenciada de filmes e vídeos em três níveis essenciais, e que visam permitir ao utilizador: 1) ter uma visão geral e potencialmente agregada sobre as colecções de filmes e vídeos; 2) individualizá-los nas colecções; e 3) visualizar e navegar pelos seus conteúdos. Em primeiro lugar, é possível ter uma visão geral de quantidades de filmes e vídeos filmados numa determinada zona geográfica e num dado momento, ver as suas trajectórias e velocidades e detalhar o critério de interesse, escolhendo uma trajectória em particular. Em segundo lugar, é possível escolher e aproximar essa trajectória de interesse obtendo detalhe e aceder a mais propriedades tais como cores, sons, e palavras proferidas; conhecer percursos vizinhos; e visionar o filme. Por fim, podem detalhar-se as propriedades dos conteúdos dos vídeos escolhidos e relacionar-

se palavras proferidas, eventos sonoros, estado de espírito do áudio, emoções sentidas, emoções nas palavras proferidas, cores dominantes, quantidade de movimento e *frames* das cenas, em sincronia com o vídeo a correr.

Em trabalho futuro temos a intenção de refinar estes estudos de design. Uma avaliação dar-nos-á uma ideia mais clara sobre a eficácia e preferências do utilizador em cada visualização e apontará direcções a explorar para melhoramentos mais sustentados.

O nosso foco será o desenvolvimento de visualizações interactivas eficazes do ponto de vista do utilizador, que possam ser enriquecidas através de técnicas de processamento de vídeo, seguindo os trabalhos MovieClouds [Gil12] e obter informação sobre a actividade de partilha e acesso dos utilizadores SightSurfers [Noronha12] [Ramalho13]. O grande objectivo será o de proporcionar aos utilizadores visualizações que lhes permitam obter visões gerais e *insights* sobre vídeos e os lugares onde foram filmados e aceder e ver os vídeos com base em critérios espacio-temporais, enriquecidos com perspectivas sobre o seu conteúdo. O cruzamento das visualizações desenvolvidas e apresentadas até este momento e a avaliação, já iniciada, dará lugar, pretende-se, a uma interface de visualização eficaz, estética e lúdica para explorar filmes e vídeos nas suas propriedades relacionadas com imagem, som e texto, explorando o tempo e a localização geográfica.

Direcções complementares incluem enriquecer a visualização de vídeos imersivos, como os que temos vindo a abordar em [Noronha12] [Ramalho13] através de vídeos panorâmicos 360° geo-referenciados, e a integração de interfaces multimodais com base na forma, velocidade e tempo [Serra14], em direcção a um acesso interactiva aos vídeos, mais natural, envolvente e flexível.

6. AGRADECIMENTOS

Este trabalho foi parcialmente suportado pela FCT através do Projecto Estratégico LaSIGE, com referência PEST-OE/EEI/UI0408/2014; o Projecto de investigação ImTV com referência UTA-Est/MAI/0010/2009; e a bolsa de doutoramento FCT dentro do programa UT Austin|Portugal com referência SFRH/BD/51798/2011.

7. REFERÊNCIAS

- [Andrienko11] G. Andrienko, N. Bak, P. Keim et al. Conceptual Framework and Taxonomy of Techniques for Analysing Movement. *Journal of Visual Languages & Computing* 22 (3) (2011) 213–32.
- [Angelewa05] J. Angelewa, Cooper, R. (2005) Last Clock. *Computer Graphics and Applications, IEEE* 25 (1): 20–23.
- [Cruz11] P. Cruz, P. Machado. Visualizing the Circulatory Problems of Lisbon. *ACM SIGGRAPH 2011* (ACM Press 2011)
- [Finsterwald12] J-M. Finsterwald, G. Grefenstette, Law-To et al. The Movie Mashup Application MoMa: Geolocalizing and Finding Movies. *Proc. of the ACM GeoMM'12* (ACM Press 2012) 15–18.

- [Gil12] N. Gil, N. Silva, Dias, et al. Going through the Clouds: Search Overviews and Browsing of Movies. *Proc. of Academic MindTrek'12*, ACM Press 2012, 158–165.
- [Hao11] J., Wang, G., Seo, B., Zimmermann, R. Keyframe Presentation for Browsing of User-Generated Videos on Map Interfaces. *Proc. ACM Multimedia*, ACM Press, 2011, 1013–16.
- [Harley87] J. B. Harley, D. Woodward. The History of Cartography. *Univ. of Chicago*, 1, Press 1987.
- [Jorge12] A. Jorge, N. Gil, T. Chambel, Time for a New Look at the Movies through Visualization. *Proc. of Artech'2012, the 6th Int. Conf. on Digital Arts*. Faro, Portugal, 2012, 269–78.
- [Jorge13] A. Jorge. Exploring Movies through Interactive Visualizations. *Proc. of the 27th Int. BCS Human Computer Interaction Conf.* London, UK, Sep 2013.
- [Kraak10], M. Kraak, J. F. Ormeling. *Cartography: Visualization of Geospatial Data*. New York, Prentice Hall 2010.
- [Levin05] G. Levin. An Informal Catalogue of Slit-Scan Video Artworks and Research. *FLONG*. 2005.
http://www.flong.com/texts/lists/slit_scan/
- [Noronha12] G. Noronha, C. Álvares, T. Chambel. Sharing and Navigating 360° Videos and Maps in Sight Surfers. *Academic MindTrek'12*, ACM Press 2012, 255–262.
- [Nunes07] M. Nunes, S. Greenberg, S. Carpendale et al. What Did I Miss? Visualizing the Past through Video Traces. *ECSCW 2007*, Springer 2007, 1–20.
- [Parent13] Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., Bogorny, V., Pelekis, N., Damiani, M., Gkoulakas-Divanis, A., Macedo, J., Pelekis, N., Theodoridis, Y., and Yan, Z., Semantic trajectories modeling and analysis. *ACM Computing Surveys*, 45(4), 2013, 1–32.
- <http://doi.org/10.1145/2501654.2501656>
- [Peterson97] M.P. Peterson. Trends in Internet Map Use. *Proc. of the 18th Int. Cartographic Conf.*. Stockholm, Sweden, 1997, 1635–1642.
- [Peuquet94] D. J. Peuquet. It's about Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems. *Annals of the Association of American Geographers* 84 (3), 1994, 441–61.
- [Peuquet01] D. J. Peuquet. Making Space for Time: Issues in Space-Time Data Representation. *GeoInformatica* 5 (1), 2001, 11–32.
- [Peuquet02] D. J. Peuquet. Representations of Space and Time. New York, The Guilford Press 2002.
- [Ramalho13] J. Ramalho, Chambel, T. Immersive 360° Mobile Video with an Emotional Perspective. *Proc. ImmersiveMe'2013*, ACM Press 2013, 35-40.
- [Rego07] A. Rego, C. Baptista, E. Silva et al. VideoLib: A Video Digital Library with Support to Spatial and Temporal Dimensions. *Proc. of ACM SAC'07*, ACM Press 2007, 1074–1078.
- [Serra14] S. Serra, A. Jorge, T. Chambel Multimodal Access to Georeferenced Mobile Video through Shape, Speed and Time. *Proc. of the 28th Int BCS Human Computer Interaction*. Southport, UK, Sep 2014.
- [Viégas04] F. Viégas, D. Boyd, D. H. Nguyen et al. Digital Artifacts for Remembering and Storytelling: Posthistory and Social Network Fragments. *Proc. of HICSS'04*, 2004.
- [Viégas09] F. Viégas, M. Wattenberg, “Flickr Flow.” *HINT.FM*, 2009.
<http://hint.fm/projects/fli>

Author Index / Índice de Autores

A

Abel Gomes	110, 133
Adérito Marcos	4
Alfredo Ferreira	88, 100, 125
Ana Jorge	24, 163
Ana Paula Afonso	71
Ana Paula Cláudio	71
Ana R. Mendes	13
António Coelho	55, 141
António Ferreira	71
Augusto Sousa	55

B

Beatriz Sousa Santos	31, 63, 80
Bruno Patrão	40

D

Daniel Mendes	88, 100, 125
Daniel Simões Lopes	13

E

Eduardo Duarte	31
Eduardo Magalhães	55
Elísio Sousa	150

G

Gonçalo Leitão	110
Gonçalo Silva	71
Gustavo Maçães	96

H

Hugo Costelha	20
Hugo da Silva	55

J

João Pinto	63
João Quintas	150
Joaquim Jorge	13, 100
Joaquim Madeira	80
Jorge Dias	47
José F. M. Morgado	133

L

Luís Almeida	47, 150
Luís Costa	141

Luis Bento	20
------------	----

M

Márcio Barata	20
Maria Beatriz Carmo	71
Maximino Bessa	55
Miguel Nixo	88
Miguel Sales Dias	125
Mirian Tavares	4

N

Nelson Alves	96
Nelson Fânzeres	96
Nuno Gonçalves	117
Nuno Sousa	96

P

Paulo Dias	31, 63
Paulo Menezes	40, 47, 150
Pedro Assunção	20
Pedro Bordonhos	31
Pedro F. Parreira	13
Pedro Miraldo	117
Pedro Pascoal	88, 125

R

Ricardo Santos	20
Rudolfo Quintas	4
Rui Coelho	96
Rui Rodrigues	55
Rui S. V. Rodrigues	133

S

Sérgio Eliseu	63
Sérgio Serra	24, 163
Samuel Pedro	40

T

Teresa Chambel	24, 163
Tiago Brito	80
Tiago Dias	117

V

Vasco Rodrigues	100
Vitor Cardoso	xvi, 149



ISBN: 978-989-20-6383-6



www.isr.uc.pt