

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



VISUALIZAÇÃO DE DADOS MÉDICOS EM
REALIDADE AUMENTADA

José Tiago Quenino Soeiro

Trabalho orientado pela Prof^ª. Doutora Ana Paula Boler Cláudio
e coorientado pela Prof^ª. Doutora Maria Beatriz Duarte Pereira do Carmo

TRABALHO DE PROJETO
MESTRADO EM INFORMÁTICA

2015

Agradecimentos

A escrita deste Relatório Final marca o final de uma fase muito importante da minha vida. Este documento é o culminar de todo um ano de trabalho que foi completamente diferente de todos os que tive até agora e certamente também de todos os que irei ter daqui em diante. Não é só o final de um mestrado, mas de todo um percurso de aprendizagem que demorou anos e que agora vai continuar mas num paradigma totalmente diferente. A aprendizagem faz-se agora de uma forma que exige muito mais autonomia, responsabilidade, sentido prático, organização, motivação, vontade de saber, de saber fazer, de fazer melhor, sempre melhor. O desafio é maior, a recompensa também.

Antes de mais, tenho de agradecer aos meus pais, por sempre me terem apoiado ao longo de todo estes anos. É graças a eles que tive todas estas oportunidades de aprendizagem, eles que sempre tiveram como objetivo bem definido possibilitarem-me estudos superiores e me apoiaram em todas as escolhas que fiz para o meu percurso académico. Porque a minha casa é sempre o sítio onde restabeleço as energias, o meu muito obrigado. Agradeço em particular à minha mãe por ter sido ajuda indispensável neste projeto, quando precisava de mais que duas mãos para fazer algum teste, tirar algumas medições ou para cozer velcro numa touca.

A marcar este ano de passagem estiveram as duas pessoas que me acolheram e me guiaram por toda esta nova aventura, as minhas orientadoras, a Professora Ana Paula Cláudio e a Professora Beatriz Carmo. Pela sua simpatia e boa disposição, por toda a sua experiência que partilharam comigo, por todo o tempo despendido para me ajudar e acompanhar, pelo espírito sempre inventivo à procura de novas soluções, por terem procurado e encontrado com prontidão um modelo de cabeça feminina e uma touca de natação tão necessitados, pela sua compreensão, por todas as ideias que deram e por terem desde o início transmitido confiança neste projeto o meu muito obrigado! Sem elas não teria sido possível, de facto, a superação deste desafio.

Agradeço também ao Professor Hugo Ferreira, que aceitou com muita alegria integrar este projeto, deu uma grande ajuda no desenho da solução e nos recebeu sempre de braços abertos e interessado nos progressos da aplicação, dando *feedback* muito útil e determinante para termos cumprido com sucesso os nossos objetivos. Por ter estabelecido

os contactos com os utilizadores necessários para testar a aplicação, por nos ter levado a demonstrar a aplicação em vários eventos e por ter sempre mostrado entusiasmo e confiança nas potencialidades deste projeto, um muito obrigado!

Quero agradecer também aos vários utilizadores de teste voluntários do IBEB e, em particular, ao Professor Mamede de Carvalho, ao Professor Alberto Leal e ao Professor Pedro Cavaleiro Miranda. Agradeço ainda à Professora Guiomar Evans do Departamento de Física da FCUL, por disponibilizar o luxímetro utilizado no teste de distância e iluminância.

Não posso deixar de agradecer à Faculdade de Ciências da Universidade de Lisboa, ao seu Departamento de Informática e ao MAS-BioISI, antigo LabMAg, por me terem proporcionado esta oportunidade, mas tenho muito que agradecer também ao Departamento de Matemática, onde me licenciiei e que contribuiu muito na minha formação e no caminho que me levou até aqui. Agradeço aos meus colegas de Matemática e Matemática Aplicada, por tudo o que passei com eles nestes cinco anos. Agradeço à Raquel Carmona e à Carolina Meireles por terem feito comigo este percurso no Mestrado em Informática e a todos os outros colegas que também me acolheram nestes dois últimos anos. Agradeço ainda aos Professores e colegas que tive ao longo de toda a minha vida.

Para terminar, mas não com menos importância, agradeço aos meus amigos. Aos amigos da FCUL, por terem estado comigo ao longo de todo este ano e acompanhado o desenvolvimento deste projeto, entusiasmando-se com os progressos, ouvindo os problemas, dando ideias ou transmitindo confiança. Por terem proporcionado os momentos de distração e diversão necessários para continuar em seguida a trabalhar com mais empenho, um grande obrigado. Aos amigos do Alentejo e, em particular, à Rute Jesus, à Marlene Oliveira e à Patrícia Santos (que viveu comigo esta aventura por Lisboa), porque, não tendo acompanhado tão de perto este projeto, sempre estiveram comigo. Aos amigos dos escuteiros, por terem compreendido quando estava ocupado, e ao escutismo em si, por continuar a oferecer-me uma diversidade de vivências e emoções fortes (principalmente neste ano tão diferente também no meu percurso escutista), por me fazer crescer e querer sempre construir um mundo melhor. Um obrigado muito especial à Tatiana Baptista e à Ana Carolina Guerreiro, pelo apoio que foram sempre e, em particular, ao longo deste ano.

À minha mãe.

Resumo

A Estimulação Magnética Transcraniana é um procedimento médico não invasivo que utiliza estímulos elétricos e magnéticos para tratar problemas neuropsiquiátricos. Nesta técnica são aplicados campos magnéticos sobre uma determinada área da cabeça do paciente para estimular os neurónios nessa área. A área a estimular depende do problema médico a resolver e a sua localização exata no cérebro varia frequentemente de pessoa para pessoa. É, pois, essencial para o médico saber a localização exata da área funcional que pretende estimular.

É neste contexto que se pretende intervir com este trabalho, concretizando uma solução que guie o médico durante a execução do procedimento. Aliou-se, então, a Realidade Aumentada e os *smartphones* para criar uma aplicação móvel que sobrepõe um modelo virtual tridimensional do cérebro sobre a cabeça do paciente. Representa-se o cérebro particionado nas suas áreas anatómicas, para indicar ao médico em que zona deve aplicar a estimulação.

Para isso, usam-se marcas semelhantes a códigos QR para rastrear a cabeça do paciente e permitir que os modelos 3D que se obtém, construídos a partir de imagens de ressonância magnética, consigam ser alinhados com a sua cabeça.

A solução apresentada inclui também um modo de Realidade Virtual, no qual o utilizador pode interagir com o modelo do cérebro. A visualização pode ser seletiva, permitindo ao utilizador escolher que áreas do cérebro pretende visualizar.

O uso do poder computacional recentemente inserido nos dispositivos móveis possibilita esta inovação, com a qual se pretende responder aos problemas reais que os médicos enfrentam ao tentar visualizar o cérebro do paciente, em particular, durante a aplicação da técnica de Estimulação Magnética Transcraniana. A solução proposta tem potencial para poder ser usada noutros contextos.

Palavras-chave: Realidade Aumentada, Realidade Virtual, Estimulação Magnética Transcraniana, cérebro, visualização, *smartphone*, aplicação Android

Abstract

The Transcranial Magnetic Stimulation is a noninvasive medical procedure that uses electric and magnetic stimulation to treat neuropsychiatric problems. In this technique, magnetic fields are applied to a particular area of the head of the patient, to stimulate the neurons in this area. The area that needs to be stimulated depends on the medical condition to be treated and its exact position is not exactly the same for every patient. It is therefore essential for the doctor to know the exact location of the functional area he wants to stimulate.

It is in this context that we aim to intervene, implementing a solution to guide the doctor during this procedure. We combined Augmented Reality and smartphones to create a mobile application that overlaps a three-dimensional virtual model of the brain over the patient's head. It represents the brain partitioned in anatomical areas, in order to guide the doctor in the procedure.

For this purpose we use markers similar to QR codes to track the patient's head, enabling the correct alignment of the 3D model of the brain with the head. The virtual model of the brain is constructed from magnetic resonance images.

The proposed solution also includes a Virtual Reality mode in which the user can interact with the brain model. Besides that, we add a functionality of selective visualization, allowing the user to choose which areas of the brain he wants to visualize.

The use of the computational power recently inserted in today's mobile devices enables this innovation, which is intended to respond to the real problems that doctors face when trying to visualize the patient's brain, with uses, in particular, in this technique of Transcranial Magnetic Stimulation. The proposed solution can potentially be used in other contexts.

Keywords: Augmented Reality, Virtual Reality, Transcranial Magnetic Stimulation, brain, visualization, smartphone, Android app

Conteúdo

Capítulo 1	Introdução	1
1.1	Motivação	1
1.2	Objetivos	2
1.3	Contribuições	4
1.4	Estrutura do Documento	6
Capítulo 2	Metodologia e Planeamento	9
2.1	Metodologia	9
2.2	Recursos e Planeamento	10
2.2.1	Recursos	11
2.2.2	Plano de trabalhos	12
2.2.3	Desvios do plano de trabalhos inicial.....	13
Capítulo 3	Conceitos Base e Trabalho Relacionado	15
3.1	Realidade Aumentada e Virtual	15
3.2	Estimulação Magnética Transcraniana	17
3.3	Aplicações da Realidade Aumentada na Medicina.....	19
3.4	Conclusão.....	23
Capítulo 4	Análise de Requisitos e Trabalho Inicial	25
4.1	Análise de Requisitos.....	25
4.1.1	Definição dos <i>Stakeholders</i>	25
4.1.2	Requisitos Funcionais	26
4.1.3	Requisitos Não-funcionais	27
4.2	Trabalho Inicial.....	28
4.2.1	Pesquisa efetuada	28
4.2.2	Escolha do equipamento.....	28
4.2.3	Escolha da ferramenta de RA.....	29
4.2.4	Familiarização com o <i>software</i> escolhido	30
4.3	Conclusão.....	31
Capítulo 5	Aplicação <i>Brain AR/VR</i>	33

5.1	Descrição da Aplicação	33
5.2	Arquitetura da Solução	37
5.3	<i>Tracking</i>	38
5.3.1	Marcas e seus referenciais.....	39
5.3.2	Ficheiro de configuração do <i>tracking</i>	42
5.3.3	Protótipo inicial da aplicação	44
5.3.4	Problemas com o <i>tracking</i> de marcas.....	46
5.3.5	Estudo das configurações de marcas	48
5.3.6	Calibração das marcas	51
5.3.7	Teste de distância e iluminância.....	54
5.3.8	Teste de uso de sensores.....	57
5.4	Visualização, Perceção de Profundidade e Estereoscopia	59
5.4.1	Perceção de Profundidade	60
5.4.2	Estereoscopia.....	63
5.5	Modo de Realidade Virtual.....	65
5.6	Visualização Seletiva	66
5.7	Aquisição e Processamento dos Modelos do Cérebro	67
5.7.1	Obtenção dos modelos	67
5.7.2	Modelos usados e sua organização.....	70
5.7.3	Ficheiro de texto de definição das geometrias	72
5.7.4	Teste de Carga.....	75
5.7.5	Processamento dos modelos com o Blender Python.....	80
5.8	Introdução de informação na aplicação	81
5.9	Conclusão.....	83
Capítulo 6	Avaliação da aplicação	85
6.1	Metodologia	85
6.2	Perfil dos Utilizadores	86
6.3	Análise dos Resultados	87
6.3.1	Modo de RA	88
6.3.2	Modo de RV	93

6.3.3	Visualização Seletiva	95
6.3.4	Apreciações Finais	97
Capítulo 7	Conclusões e Trabalho Futuro	101
7.1	Discussão	102
7.2	Trabalho Futuro	105
7.3	Conclusão.....	107
Anexo A:	Manual de Utilizador	109
Anexo B:	Arquitetura da aplicação e fluxo de informação associado	119
Anexo C:	Cálculos associados à calibração das marcas	121
Anexo D:	Estrutura do cérebro segundo vários atlas	125
Anexo E:	Divisão das áreas do atlas DK pelos 4 lobos	129
Anexo F:	Exemplo de Ficheiro de Definição das Geometrias	131
Anexo G:	Script Python para decimar modelos no Blender	133
Anexo H:	Guião de Testes de Utilizador	135
Anexo I:	Fotografias dos Testes de Utilizador.....	143
Bibliografia	145

Lista de Figuras

Figura 2.1 - Esquema conceptual do Processo Iterativo e Incremental [9].....	9
Figura 3.1 - "Continuum Realidade-Virtualidade" [11].....	15
Figura 3.2 - Triângulo da RV: Imersão, Interação, Imaginação [13].....	17
Figura 3.3 - Funcionamento da técnica de EMT, com o coil a enviar impulsos eletromagnéticos para a região do cérebro sob ele [15]	18
Figura 3.4 - Aparelho de estimulação real existente no Hospital Santa Maria	18
Figura 3.5 - Google Glass aplicados à Medicina [21][22].....	20
Figura 3.6 - Imagens médicas realistas mostrando estruturas internas do corpo [24]	20
.....	
Figura 3.7 – Esquema funcional do sistema de realidade mista proposto por Ferrari et al.[25].....	21
Figura 3.8 - Aplicação Mind Mirror, para visualizar em tempo real a atividade cerebral [27].....	21
Figura 3.9 - (a) Vista real do fígado, (b) visualização da vesícula biliar, (c) visualização de estruturas de risco, (d) visualização de ambas as componentes [30]22	
Figura 5.1 - Interface inicial da aplicação, no modo de RA e com os botões de escolha de modo e para o menu de opções visíveis.....	34
Figura 5.2 - Indicadores de calibração	34
Figura 5.3 - Glifo do aparelho de estimulação nas duas representações: Arrow (esquerda) e Coil (direita).....	35
Figura 5.4 - Ecrã do modo de RV, com todos os botões da interface visíveis.....	36
Figura 5.5 - Ecrã do menu de opções	36
Figura 5.6 - Arquitetura da aplicação.....	38
Figura 5.7 – Marca de 60 mm com ID 5 e o sentido dos eixos ‘x’ e ‘y’ assinalados	40
.....	
Figura 5.8 – Esquerda: Estrutura global do ficheiro de configuração de tracking com 5 marcas; Em cima à direita: Estrutura de um elemento <SensorCOS>; Em baixo à direita: Estrutura de um elemento <SensorSource>	44
Figura 5.9 – Os cinco primeiros ID Markers fornecidos pelo Metaio	45
Figura 5.10 - O modelo físico de cabeça feminina durante o processo de medição, com as marcas colocadas na touca	45

Figura 5.11 - Touca com as marcas só em papel (esquerda) e em papel coladas em cartão (direita).....	47
Figura 5.12 - a) 5 marcas de 15 mm; b) 5 marcas de 30 mm; c) 5 marcas de 45 mm; d) 9 marcas de 15 mm; e) 9 marcas de 30 mm; f) 15 marcas de 15 mm.....	49
Figura 5.13 – Interface do modo de RA durante o processo de calibração (esquerda) e quando todas as marcas ficam calibradas (direita)	53
Figura 5.14 – Luxímetro usado no teste de distância e luminância	54
Figura 5.15 - Modelo virtual do cérebro com várias áreas distinguidas pela cor, com uma cor mais distintiva para a área a estimular.....	60
Figura 5.16 - Glifo do aparelho de estimulação nas duas representações: Arrow (esquerda) e Coil (direita).....	62
Figura 5.17 – Ecrã do modo de RA com o glifo do aparelho de estimulação das várias cores, consoante a distância deste ao modelo do cérebro: distantes (esquerda), a distância intermédia (centro) e próximos (direita)	62
Figura 5.18 - Visualização de um modelo 3D com o ecrã dividido ao meio.....	64
Figura 5.19 - Visualização de um modelo 3D com fundo colorido: opaco (esquerda); com transparência (direita)	65
Figura 5.20 - Ecrã do modo de RV, com todos os botões da interface visíveis....	66
Figura 5.21 - Ecrã do menu de opções (esquerda) e modo de RV com visualização seletiva (direita)	67
Figura 5.22 - Imagens ilustrativas das parcelizações de acordo com os três atlas e nas três variantes (versão modificada da imagem apresentada na página do projeto Brainder [38])	71
Figura 5.23 - Excerto de um ficheiro de definição das geometrias usado durante o desenvolvimento do projeto.....	74
Figura 5.24 - Ecrã do modo de RV apresentando o cérebro colorido de acordo com o ficheiro de definição de geometrias exemplificado.....	75
Figura 5.25 – Relação entre o número de polígonos a desenhar e o número de fps do rendering, relativamente a três smartphones Android diferentes [5].....	79
Figura 5.26 – Relação entre o número de polígonos a desenhar e o número de fps do rendering, relativamente a três formatos de ficheiro diferentes para os modelos, sempre num Samsung Galaxy S4 [5]	79
Figura 6.1 – Gráficos da distribuição dos utilizadores de teste por níveis autoatribuídos numa escala de 1 a 5 em relação a: experiência de utilização de	

smartphones, experiência de utilização de ecrãs táteis e conhecimentos de Realidade Aumentada ou Virtual	87
Figura 6.2 - Gráficos da distribuição das respostas relativamente ao processo de calibração em três componentes: facilidade, duração e utilidade dos indicadores.....	89
Figura 6.3 – Gráfico da distribuição das respostas relativamente ao alinhamento do modelo virtual no modo de RA	89
Figura 6.4 - Gráficos da distribuição das respostas relativamente à utilidade da indicação colorida da distância e intuitividade do código de cores usado	90
Figura 6.5 – Comparação entre os gráficos da distribuição das respostas relativamente à perceção de profundidade, representação visual do aparelho de estimulação e a qualidade da experiência de estimulação para os dois glifos alternativos do aparelho de estimulação: “Arrow” e “Coil”	92
Figura 6.6 – Gráfico da distribuição das respostas relativamente à apreciação global do modo de RA.....	93
Figura 6.7 – Gráficos da distribuição das respostas relativamente à facilidade de ajustar o tamanho do modelo, intuitividade do gesto para ajustar o tamanho, facilidade de colocar o modelo numa orientação desejada sem recorrer às vistas predefinidas, intuitividade do gesto para rodar o modelo e utilidade das vistas predefinidas	94
Figura 6.8 - Gráfico da distribuição das respostas relativamente à apreciação global do modo de RV.....	95
Figura 6.9 - Gráficos da distribuição das respostas relativamente à organização do menu de opções, facilidade de alterar as áreas visíveis e utilidade da funcionalidade de Visualização Seletiva.....	96
Figura 6.10 – Gráfico da distribuição das respostas relativamente à apreciação global da funcionalidade de Visualização Seletiva	97
Figura 6.11 – Gráficos da distribuição das respostas relativamente às apreciações globais das funcionalidades da aplicação, capacidades de visualização fornecidas, interface de utilizador e utilidade/relevância da aplicação	98
Figura 6.12 – Gráficos da distribuição das respostas relativamente às apreciações globais do modo de RA, modo de RV e funcionalidade de Visualização Seletiva.....	99

Lista de Tabelas

Tabela 1 - Resultados do estudo de configuração das marcas	49
Tabela 2 - Gamas de iluminância de diversas condições ambiente	55
Tabela 3 - Resultados do teste de distância/iluminância.....	56
Tabela 4 - Resultados do teste de carga	78

Lista de Acrónimos e Siglas

ADT	<i>Android Developer Tools</i>
API	<i>Application Programming Interface</i>
COS	<i>CoOrdinate System</i>
DI	Departamento de Informática
DICOM	<i>Digital Imaging and COmmunications in Medicine</i>
EMT	Estimulação Magnética Transcraniana
FCUL	Faculdade de Ciências da Universidade de Lisboa
HMD	<i>Head Mounted Display</i>
HTML	<i>HyperText Markup Language</i>
IBEB	Instituto de Biofísica e Engenharia Biomédica
I&D	Investigação e Desenvolvimento
LabMAg	Laboratório de Modelação de Agentes
MAS-BioISI	Agent and Systems Modelling – Biosystems & Integrative Sciences Institute
MATLAB	<i>MATrix LABoratory</i>
MI	Mestrado em Informática
NIFTI	<i>Neuroimaging Informatics Technology Initiative</i>
PEI	Projeto de Engenharia Informática
QR	<i>Quick Response</i>
RA	Realidade Aumentada
RV	Realidade Virtual
SDK	<i>Software Development Kit</i>
XML	<i>eXtensible Markup Language</i>

Capítulo 1

Introdução

Este trabalho enquadra-se na unidade curricular Projeto de Engenharia Informática (PEI), que está integrada no 2º ano do Mestrado em Informática (MI) da Faculdade de Ciências da Universidade de Lisboa (FCUL). O grupo de I&D que suporta o desenvolvimento deste projeto é o MAS (Agent and Systems Modelling) do BioISI (Biosystems & Integrative Sciences Institute) [1]. Este grupo dá continuidade ao LabMAG (Laboratório de Modelação de Agentes), uma das unidades de I&D do Departamento de Informática (DI) da FCUL.

Este trabalho está a ser desenvolvido em colaboração com o Professor Hugo Ferreira do Instituto de Biofísica e Engenharia Biomédica (IBEB/FCUL).

1.1 Motivação

A Estimulação Magnética Transcraniana (EMT) é um procedimento médico não invasivo que utiliza estímulos elétricos e magnéticos para tratar problemas neurológicos e psiquiátricos. É normalmente usada para curar depressões, esquizofrenia, dor crónica e as suas aplicações continuam a aumentar [2]. Nesta técnica aplicam-se campos magnéticos sobre a cabeça do paciente, numa área específica em função do problema que se pretende resolver, para estimular os neurónios dessa área e devolver à região o seu funcionamento normal.

O médico que aplica esta técnica sabe qual a área do cérebro que pretende estimular e sabe aproximadamente onde essa área se situa na estrutura global da cabeça do paciente, mas cada cérebro é diferente e o nível de desenvolvimento de cada região varia, pelo que o médico pode estar a estimular a área errada, o que leva à não existência de melhorias no paciente e possivelmente até a efeitos indesejados. É então muito importante para o médico saber exatamente sobre que zona deve aplicar o estímulo.

A possibilidade de visualizar o cérebro do paciente em simultâneo com a intervenção pode aumentar significativamente a segurança e eficiência do tratamento.

Normalmente, os médicos recorrem às imagens obtidas através de ressonâncias magnéticas e outros métodos de imagiologia, mas visualizam-nas muitas vezes em 2D e têm de sobrepor as imagens para conseguir obter uma melhor noção da estrutura 3D do cérebro. Além disso, é sempre um processo de visualização fisicamente afastado do paciente.

É para este problema, não apenas da EMT mas de muitos procedimentos médicos, que a Realidade Aumentada (RA) veio trazer soluções. A RA permite aos médicos visualizar modelos 3D dos órgãos do paciente sobre o próprio, como se estivesse efetivamente a vê-los. É uma forma de visualização que proporciona cada vez melhores resultados e as suas aplicações, tanto a nível médico como de entretenimento e na indústria, têm vindo a aumentar cada vez mais.

É neste contexto que surge este projeto. Pretende-se aplicar a RA à Medicina e, mais concretamente, à EMT, permitindo ao médico que executa a técnica visualizar o modelo do cérebro do paciente sobre a cabeça do mesmo, com informação sobre a área que deve ser estimulada, para facilitar e trazer mais segurança e eficácia a esta técnica. Uma das mais-valias do projeto é também a utilização de *smartphones* como ferramenta de computação e visualização das imagens de RA, pois são dispositivos muito comuns e acessíveis, com os quais os médicos facilmente trabalharão, e que possuem já suficiente poder computacional para desempenhar as tarefas pretendidas.

1.2 Objetivos

O objetivo geral deste trabalho foi o de conceber e implementar uma aplicação para *smartphone* que possibilite uma visualização em RA do cérebro de pacientes sobre a cabeça dos mesmos, o que é possível graças ao uso de marcas colocadas na cabeça do paciente que o *software* consegue reconhecer. Para além disto, inclui-se também um modo de Realidade Virtual (RV), em que o utilizador pode interagir de forma táctil com o modelo virtual do cérebro, rodando o modelo ou aumentando/diminuindo o seu tamanho. Possibilita-se ainda uma funcionalidade de Visualização Seletiva, que permite ao utilizador filtrar que áreas do cérebro pretende visualizar, tanto no modo de RA como no modo de RV.

Os objetivos gerais definidos inicialmente para o projeto foram os seguintes:

- perceber como obter e preparar modelos 3D do cérebro a partir de imagens de ressonância magnética, para que possam ser visualizados na aplicação;
- detetar marcas colocadas na cabeça do paciente, para se saber de forma precisa qual a posição e rotação e o tamanho com que o modelo 3D deve ser desenhado;
- apresentar na aplicação, de alguma forma, um guia visual para a área (ou áreas) específica do cérebro que se quer estimular, de forma a auxiliar o médico no procedimento de EMT;
- estudar e descobrir a melhor forma de apresentar a visualização ao médico, resolvendo problemas de perceção de profundidade e avaliando o uso de estereoscopia e de um suporte ocular para o *smartphone* para que o médico possa executar a técnica com ambas as mãos livres.

As tarefas e objetivos que se pretenderam concluir ao longo deste trabalho, enumeradas em mais detalhe, foram as seguintes:

- Trabalho prévio ao desenvolvimento:
 - Fazer uma recolha e análise sobre o estado-da-arte das aplicações de RA na Medicina;
 - Estudar qual o melhor sistema operativo móvel para desenvolver a aplicação, qual o melhor *smartphone* para usar como aparelho de teste e qual o melhor suporte ocular para *smartphone* para usar durante a execução do projeto;
 - Estudar qual a melhor ferramenta de *software* para usar na criação da aplicação;
 - Perceber qual a melhor forma de obter e preparar os modelos virtuais 3D do cérebro para visualização na aplicação;
 - Perceber qual a melhor forma de fazer o *tracking* da cabeça do paciente;
 - Estudar e experimentar o *software* escolhido para o desenvolvimento da aplicação, para perceber e fazer um plano mais detalhado de como desenvolver a aplicação;

- Desenvolvimento da aplicação:
 - Implementar o modo de Realidade Aumentada, com base em todo o planeamento feito anteriormente;
 - Desenhar duas representações alternativas para o aparelho de estimulação, com o intuito de melhorar a perceção de profundidade no modo de RA e melhor guiar o médico no procedimento de EMT;
 - Implementar um modo de Realidade Virtual, para interação tátil com o modelo virtual, e desenhar a interface de utilizador associada, bem como forma de alternar entre os dois modos (RA e RV);
 - Implementar uma funcionalidade de Visualização Seletiva das áreas do cérebro, com utilização de um ficheiro de texto para fornecer à aplicação informações sobre as várias áreas do cérebro;
 - Criar um menu de opções que permita selecionar qual a representação a usar para o aparelho de estimulação e também escolher que áreas do cérebro visualizar no modelo virtual;
 - Criar um *script* para autonomizar o processo de decimação (redução do número de polígonos) dos modelos virtuais obtidos;
- Testar a aplicação, avaliando as funcionalidades disponibilizadas, a interface de utilizador, a qualidade do *tracking* e da visualização, fazendo melhorias na aplicação até chegar a um resultado desejável, dentro das limitações existentes;
- Escrever um artigo científico que seja aceite numa conferência na área;
- Escrever o Relatório Final.

1.3 Contribuições

A principal contribuição deste trabalho foi a criação de uma aplicação para dispositivos móveis Android, designada por *Brain AR/VR*, que fornece capacidades de visualização de modelos virtuais 3D do cérebro de pacientes em Realidade Aumentada e Virtual. A aplicação, no seu modo de RA, foi pensada para uso como guia durante a aplicação da técnica médica de Estimulação Magnética Transcraniana, sendo capaz de alinhar um modelo virtual do cérebro do paciente com a imagem da cabeça captada pela câmara do dispositivo móvel, desde que este esteja a usar uma touca com marcas fiduciais

corretamente posicionada. Representações para o aparelho de estimulação da técnica de EMT, também na forma de modelos virtuais 3D, foram criadas e permitem melhorar a percepção de profundidade da experiência de RA, sendo que o utilizador pode escolher qual das duas representações criadas quer visualizar. O modo de RV apresenta os modelos 3D num ambiente meramente virtual que possibilita a interação tátil por parte dos utilizadores, que podem rodar ou alterar a escala dos modelos, podendo ainda selecionar uma de seis vistas predefinidas. Existe ainda uma funcionalidade de Visualização Seletiva que permite ao utilizador escolher, num menu de opções, quais as áreas do cérebro que a aplicação deve representar, tanto no modo de RA como no modo de RV. A aplicação, totalmente funcional, foi testada por utilizadores peritos das áreas da Engenharia Biomédica, da Medicina e da Biofísica, envolvidos em estudos do cérebro humano.

Para o desenvolvimento da aplicação foram usadas as *Android Developer Tools* [3] integradas no ambiente de desenvolvimento Eclipse [4]. Para fazer a deteção das marcas e representar os modelos virtuais foi adicionada ao projeto Android a biblioteca de classes Metaio SDK, disponibilizada pela empresa Metaio [5]. A criação dos modelos virtuais que representam o aparelho de estimulação foi feita com recurso ao *software* de modelação 3D Blender [6], que também foi usado para reduzir o número de polígonos dos modelos virtuais do cérebro que são obtidos a partir da ressonância magnética.

Para além da aplicação e da escrita deste relatório final, que descreve todo o trabalho realizado e as várias abordagens tentadas e estudos efetuados, ainda foi escrito um artigo científico [7]. Este *paper* foi submetido e aceite para ser apresentado numa conferência internacional, a EMBC 2015 [8]. Esta conferência abrange não só a área da Medicina como se foca em soluções tecnológicas inovadoras para resolver problemas dessa área. A referência do artigo é a seguinte:

José Soeiro, Ana Paula Cláudio, Maria Beatriz Carmo, Hugo Alexandre Ferreira, *Visualizing the Brain on a Mixed Reality Smartphone Application*, Artigo aceite para a conferência EMBC 2015, 37ª Conferência Internacional Anual da “IEE Engineering in Medicine and Biology Society”, Milão, Itália, 25 a 29 de Agosto de 2015

Durante os últimos meses de desenvolvimento da aplicação, surgiram algumas oportunidades para a apresentar ao público, fazendo demonstrações do uso da mesma. Estas oportunidades passaram por:

- uma presença em exposição e demonstração a alunos do Ensino Secundário no Dia Aberto 2015 da FCUL;
- uma apresentação e demonstração a alunos de Mestrado do Departamento de Informática da FCUL no âmbito da cadeira de Animação e Ambientes Virtuais;
- filmagens da aplicação em funcionamento para um vídeo de divulgação da FCUL;
- uma demonstração numa reunião com elementos de um projeto de criação de um museu interativo, a Universidade dos Valores, em Mafra;
- uma presença em exposição e demonstração na FCUL inserida no ciclo de palestras mensais “Noites de Ciências, Noites de Luz”, em comemoração do Ano Internacional da Luz (2015);
- uma presença em exposição e demonstração no Salão Nobre do Instituto Superior Técnico no âmbito do *demo day* da *Lisbon Machine Learning School (LXMLS)* 2015.

1.4 Estrutura do Documento

Este documento encontra-se organizado da seguinte maneira:

- Capítulo 2 – Metodologia e Planeamento: Neste capítulo expõe-se a metodologia utilizada durante o desenvolvimento do projeto e o planeamento do mesmo, sendo apresentados os recursos, o plano de trabalhos inicial e os desvios que se efetuaram ao plano.
- Capítulo 3 – Conceitos Base e Trabalho Relacionado: Neste capítulo apresentam-se conceitos base necessários e dá-se um panorama do estado-da-arte na área.
- Capítulo 4 – Análise de Requisitos e Trabalho Inicial: Neste capítulo faz-se a análise de requisitos do projeto e descreve-se o trabalho realizado prévio ao desenvolvimento da aplicação.

- Capítulo 5 – Aplicação *Brain AR/VR*: Neste capítulo descreve-se a aplicação e todos os seus componentes.
- Capítulo 6 – Avaliação da aplicação: Neste capítulo aborda-se o trabalho realizado de avaliação da aplicação com utilizadores, apresentando-se a metodologia usada e os resultados obtidos.
- Capítulo 7 – Conclusões e Trabalho Futuro: Neste capítulo analisa-se a solução desenvolvida, as dificuldades que surgiram e descrevem-se ideias para trabalho futuro.

Capítulo 2

Metodologia e Planeamento

Neste capítulo expõe-se a metodologia utilizada durante o desenvolvimento do projeto bem como o planeamento do mesmo, sendo apresentados os recursos do projeto, o plano de trabalhos inicial e feitas considerações em relação aos desvios que se efetuaram a este plano de trabalhos.

2.1 Metodologia

Para o desenvolvimento deste projeto foi escolhido o Processo Iterativo e Incremental. Neste processo, existe uma fase inicial de planeamento a que se segue uma fase de desenvolvimento sucessivo de diversas iterações da aplicação, cada vez mais completas, até se chegar a um produto final, como esquematizado na Figura 2.1.

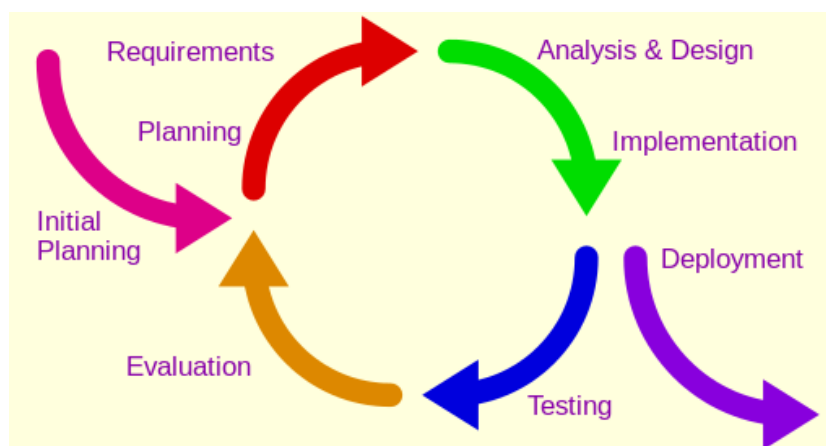


Figura 2.1 - Esquema conceptual do Processo Iterativo e Incremental [9]

Inicialmente, foi feito um trabalho de pesquisa sobre o estado-da-arte na área, fazendo o levantamento de problemas interessantes discutidos por diversos artigos que eram pertinentes para o desenvolvimento da aplicação que se pretendia criar. Com o auxílio das soluções apresentadas por estes artigos, foi simultaneamente sendo trabalhado o plano de como a aplicação de RA iria funcionar, o que implicava pensar em como obter e tratar os modelos, como fazer o *tracking* da cabeça do paciente e alinhar os modelos em

função disso, como visualizar os modelos e como introduzir toda a informação e ficheiros necessários no dispositivo móvel para a aplicação conseguir ler. À medida que todos estes procedimentos iam sendo pensados, foram escolhidas as ferramentas de *software* a usar no desenvolvimento e também que dispositivo móvel adquirir.

No final de todo este processo de planeamento, desenvolveu-se um protótipo mostrando a funcionalidade básica da aplicação, provando que o conceito era possível de ser concretizado. A partir daí estabeleceu-se um plano de trabalhos com a lista de todos os elementos chave que era preciso desenvolver ou melhorar na aplicação. Esta lista de tarefas acompanhou todo o processo de implementação da aplicação, sendo constantemente avaliada e alterada em função dos avanços que se iam verificando no trabalho realizado.

Começou assim a fase iterativa, em que com muita regularidade se concluía uma nova pequena iteração da aplicação e esta iteração era avaliada entre todos os elementos da equipa. Os problemas que tinham surgido durante o desenvolvimento eram discutidos e eram propostas soluções, ou então tentativas de abordagens alternativas, que se acrescentavam à lista de tarefas. Era analisado o trabalho desenvolvido desde a iteração anterior, para se perceber o que já se podia dar por concluído da lista de tarefas e o que ainda faltava melhorar, ou o que simplesmente não resultava e tinha de ser alterado ou excluído. Também foram surgindo novas ideias que foram adicionadas à lista de tarefas, que esteve em constante evolução durante todo o processo.

O plano de trabalhos inicial do projeto foi tido sempre como base e o tempo que restava para o fim do projeto foi avaliado ao longo das várias iterações e tido em conta no planeamento que se fazia para as futuras iterações, de modo a tentar garantir ao máximo o cumprimento dos prazos definidos.

2.2 Recursos e Planeamento

Nas secções seguintes descrevem-se os recursos do projeto, a calendarização e a análise dos desvios ao plano de trabalhos inicial.

2.2.1 Recursos

Nesta secção são descritos todos os recursos utilizados ao longo do projeto.

Pessoas envolvidas:

As pessoas envolvidas neste projeto são as descritas como *stakeholders* não-utilizadores do projeto (ver secção 4.1.1).

Hardware:

O *hardware* utilizado no desenvolvimento deste projeto é:

- *Smartphone* Samsung Galaxy S4 GT-I9505
 - Sistema Operativo: Android 4.4.2 Kit-Kat
 - Processador: Octa-core 1.6GHz
 - Memória RAM: 2GB
 - GPU: PowerVR SGX 544MP3
 - Câmara: 13MP
 - Resolução de Ecrã: 1080x1920px

Software:

O *software* utilizado no desenvolvimento deste projeto é:

- Eclipse (versão Juno) com as ADT (Android Developer Tools)
- Metaio SDK (Software Development Kit) – versões 5.5.2 e 6.0.2
- Blender 2.71

Outros materiais:

Para além dos equipamentos referidos acima, foram também usados os seguintes materiais:

- Cabeça feminina de material plástico
- *ID Markers* do Metaio, impressos em papel e colados sobre cartolina/cartão
- Touca de natação sobre a qual se fixam as marcas e que se coloca na cabeça
- Régua de 20 cm

2.2.2 Plano de trabalhos

O plano de trabalhos inicial era o seguinte:

1 Outubro – 30 Novembro:

- Identificação e análise do problema;
- Levantamento do estado-da-arte;
- Familiarização com as ferramentas a utilizar;
- Planeamento técnico do procedimento;
- Desenvolvimento de um protótipo;
- Escrita do Relatório Preliminar.

1 Dezembro – 31 Maio:

- Definição do procedimento de preparação dos modelos 3D;
- Melhoramento do *tracking*;
- Criação dos restantes elementos da visualização;
- Estudo sobre a utilização de estereoscopia e do suporte ocular para *smartphone*, com conseqüente melhoramento da visualização;
- Criação da interface de utilizador da aplicação;
- Testes da aplicação, avaliando a interface de utilizador, a facilidade de inserção dos modelos 3D e ficheiros de configuração e a qualidade do *tracking* e da visualização;
- Melhorar e corrigir a aplicação em função dos resultados dos testes.

1 Junho – 30 Junho:

- Revisão e especificação final do procedimento;
- Últimos ajustes à aplicação;
- Escrita do Relatório Final.

2.2.3 Desvios do plano de trabalhos inicial

Temporalmente, o decorrer do projeto não sofreu desvios significativos em relação ao plano de trabalhos, tendo a primeira fase (meses de Outubro e Novembro) sido cumprida em termos de tarefas com exatidão. Na segunda fase (de Dezembro até Maio), porém, à medida que a aplicação ia sendo desenvolvida, algumas das ideias originais tiveram de ser adaptadas e novas ideias surgiram também para ser implementadas no âmbito do projeto.

Em particular, o estudo que se fez sobre a utilização de estereoscopia e do suporte para *smartphone* não deu os resultados pretendidos, não tendo sido possível chegar a uma solução de visualização estereoscópica agradável. No que diz respeito ao *tracking* e ao alinhamento resultante dos modelos, a solução inicial de medir as relações de todas as marcas com um mesmo referencial central revelou-se bastante pouco prática e uma nova solução de calibração das marcas pela aplicação teve de ser implementada. Ainda assim, o método de *tracking* com marcas revelava algumas imperfeições, pelo que se usou algum tempo do projeto a testar uma nova abordagem que usava a informação fornecida pelos sensores do telemóvel para fazer os cálculos necessários ao alinhamento do modelo virtual do cérebro com a imagem da cabeça captada pela câmara. Esta solução obteve, à partida, resultados piores e, como o tempo para o desenvolvimento do projeto estava já bastante limitado, os esforços foram concentrados novamente na abordagem de uso das marcas calibradas. Também alguns testes de desempenho que se fizeram ao longo do desenvolvimento, estudando a influência de diversos fatores, acabaram por fazer com que o período de desenvolvimento se estendesse para além do previsto.

Para além disto, ao longo do desenvolvimento do projeto percebeu-se a utilidade de acrescentar mais alguma funcionalidade à aplicação, em particular um modo de Realidade Virtual para interação tátil com os modelos 3D e ainda a capacidade de visualização seletiva de áreas do cérebro. Estas duas funcionalidades foram então implementadas na aplicação, na altura em que a interface de utilizador começou a ser trabalhada. Outro elemento adicional que não constava do plano de trabalhos inicial foi a escrita de um artigo científico [7].

Depois de a aplicação atingir uma iteração quase final foram então pensados os testes de utilizador para avaliar a aplicação, tendo sido criado um guião de entrevista. Este

trabalho foi feito já no final de Maio, portanto ligeiramente atrasado, devido a todas as alterações ao plano referidas acima. Os testes de utilizador começaram ainda em Maio mas estenderam-se pelo mês de Junho, ao mesmo tempo que foi escrito o Relatório Final. Como era importante fazer os testes de utilizador com pessoas da área, em particular com médicos que tivessem experiência na aplicação da técnica de EMT, a marcação dos testes de utilizador ficou dependente das agendas dos médicos que se disponibilizaram para testar a aplicação, o que fez com que a fase de testes só ficasse concluída no princípio de Julho. Assim, a conclusão do Relatório Final só ocorreu também em Julho, ligeiramente atrasada em relação ao plano de trabalhos. Não houve, em particular, tempo para realizar muitas alterações à aplicação em função dos resultados dos testes, que foram positivos. Ainda assim, foram dadas diversas sugestões para mudanças ou adição de funcionalidades na aplicação, o que será deixado para trabalho futuro.

Capítulo 3

Conceitos Base e Trabalho Relacionado

Neste capítulo apresentam-se conceitos base necessários para compreender o projeto, em particular os conceitos de Realidade Aumentada e Virtual e de Estimulação Magnética Transcraniana, e mostram-se exemplos de outros projetos e usos da Realidade Aumentada na Medicina, tentando dar-se um panorama do estado-da-arte na área.

3.1 Realidade Aumentada e Virtual

Realidade Aumentada (RA) e Realidade Virtual (RV) são duas abordagens diferentes de visualização ou interação homem-máquina, pertencentes ao “*Continuum da Realidade-Virtualidade*” (Figura 3.1). Este “*continuum*” é uma escala que varia desde ambientes puramente reais até ambientes completamente virtuais, abrangendo todas as técnicas intermédias de Realidade Mista, em que elementos virtuais e reais são misturados num só ambiente ou visualização. [10]

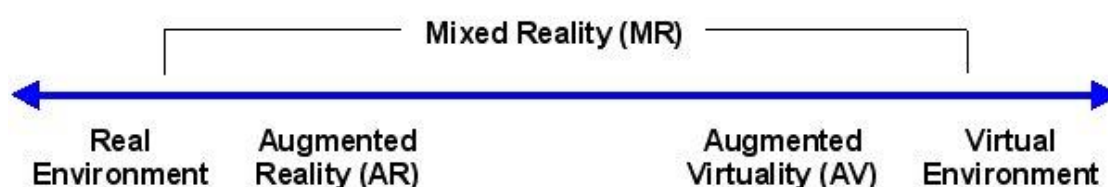


Figura 3.1 - "Continuum Realidade-Virtualidade" [11]

Assim, a RA é uma abordagem em que elementos virtuais aumentam (dão mais informação) o mundo real, enquanto na Virtualidade Aumentada são elementos reais que aumentam um ambiente maioritariamente virtual. Na RV, por sua vez, toda a experiência é virtual, tentando-se usualmente que haja precisamente uma imersão num ambiente digital e afastamento do contexto real em que se está.

Em RA o utilizador consegue ver o mundo real, diretamente ou através de câmaras; um computador que recebe informação do mundo real gera elementos digitais que mostra

ao utilizador recorrendo, por exemplo, a óculos *see-through* (RA ótica) ou incorporando esses elementos virtuais na imagem da câmara que o utilizador está a ver, por exemplo no ecrã de um *smartphone* (RA de vídeo). Para além de um processador e de um *display*, um dos componentes mais importantes de um sistema de RA é uma forma de fazer o *tracking* (rastreo ou seguimento) de elementos reais, necessário na maior parte das aplicações, em que os elementos reais devem influenciar o posicionamento e/ou o tipo dos elementos digitais. O *tracking* pode basear-se em diferentes processos:

- *tracking* ótico, em que é preciso fazer o reconhecimento da imagem do que é captado do mundo real através de câmaras. O reconhecimento de imagem pode consistir na deteção de marcas próprias (marcas fiduciais), *QR codes* ou imagens de objetos específicos (marcas naturais), mas também pode ser usada a imagem da câmara para se fazer um mapa 2D ou 3D de pontos do mundo real e a partir dos pontos reconhecidos em cada instante na imagem da câmara calcular-se a posição do utilizador.
- *Tracking* não-ótico, consistindo no uso de GPS, por exemplo, para obter a posição do utilizador em cada instante, em combinação com o uso de sensores inerciais, permitindo que a orientação e movimentos feitos pelo utilizador sejam interpretados e que os elementos virtuais possam ser posicionados em conformidade.

Qualquer forma de *tracking* exige um elevado poder de processamento e *software* ou *hardware* específico, sendo que em alguns dos casos ainda se está a fazer progressos na tecnologia para permitir experiências de RA fluidas e fiáveis.

A RA ainda é um campo em crescimento, mas já com muitas aplicações nas mais diversas áreas. Inicialmente usada nas áreas militar, industrial e médica, começou mais recentemente a ter sucesso em aplicações comerciais e de entretenimento, graças ao aparecimento de equipamentos móveis de baixo custo mas já com capacidade para suportar experiências de RA [12].

A RV tem por base componentes muito semelhantes à RA, sendo o *tracking* da posição do utilizador essencial em grande parte das aplicações, mas, neste caso, para que

o mundo virtual se mova como se o utilizador se estivesse a mover dentro dele. Esse *tracking* é feito na maioria dos casos através de sensores de movimento e a visualização pode ser feita, por exemplo, com um *Head-Mounted Display* (HMD), um dispositivo imersivo de RV. A imersão é de facto um dos grandes objetivos das aplicações usuais de RV, juntamente com a possibilidade de interação com o mundo virtual e com o incentivo à imaginação do utilizador. Estes três componentes – imersão, interação e imaginação – formam o chamado Triângulo da Realidade Virtual: I^3 (Figura 3.2) [13].

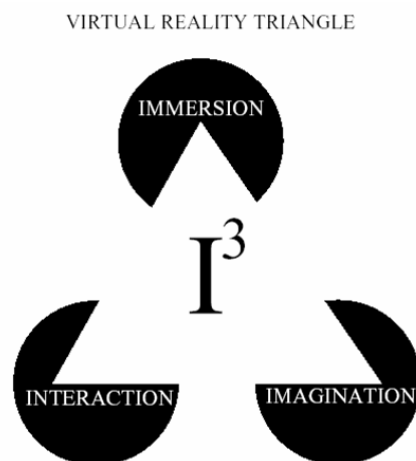


Figura 3.2 - Triângulo da RV: Imersão, Interação, Imaginação [13]

Para melhorar a imersão, muitas aplicações atuais usam *displays* estereoscópicos, conferindo ao utilizador uma melhor perceção de profundidade no ambiente virtual, e algumas utilizam também som e, mais recentemente, sistemas hápticos para introduzir informação tátil, o que é usualmente designado por *haptic feedback* em aplicações médicas, militares e em videojogos [14].

3.2 Estimulação Magnética Transcraniana

A Estimulação Magnética Transcraniana (EMT) é uma técnica não invasiva usada para estimular pequenas regiões do cérebro. Durante o procedimento, um gerador de campo magnético (“*coil*”) é colocado junto à cabeça da pessoa que está a receber o tratamento. O aparelho de estimulação produz pequenas correntes elétricas na região do cérebro diretamente sob o aparelho, através de indução eletromagnética, como esquematizado na Figura 3.3. Esta técnica pode consistir num único impulso eletromagnético ou em impulsos continuados (EMT repetitiva).

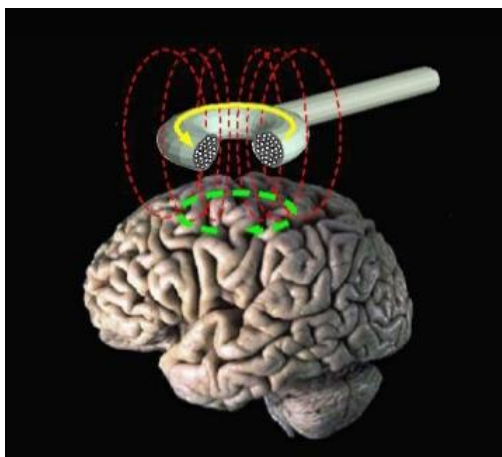


Figura 3.3 - Funcionamento da técnica de EMT, com o coil a enviar impulsos eletromagnéticos para a região do cérebro sob ele [15]

A EMT é usada, em termos de diagnóstico, para medir a conexão entre o cérebro e um músculo para avaliar danos causados por acidentes vasculares cerebrais (AVC), esclerose múltipla, esclerose lateral amiotrófica, desordens de movimento, doença do neurónio motor, entre outros problemas afetando nervos faciais e craniais ou a espinal medula. Em termos de tratamento, a EMT de um único impulso foi aprovada para uso em enxaquecas e a EMT repetitiva para uso em transtorno depressivo maior (depressão) quando este é resistente ao tratamento mais tradicional. Há indícios que esta terapia também é útil para dor neuropática, perda de funções causada por AVC e em sintomas negativos da esquizofrenia (falta de emoção, pobreza do discurso, incapacidade de sentir prazer, falta de desejo de formar relacionamentos e falta de motivação) [15][16]. Na Figura 3.4 pode ver-se um aparelho de estimulação real, utilizado no Hospital Santa Maria.



Figura 3.4 - Aparelho de estimulação real existente no Hospital Santa Maria

3.3 Aplicações da Realidade Aumentada na Medicina

Dada a possibilidade da RA enriquecer a realidade, adicionando-lhe objetos virtuais tridimensionais em tempo real, o seu potencial de aplicação no domínio do ensino da Medicina e para a preparação e facilitação de intervenções cirúrgicas torna-se facilmente compreensível. Deste modo, não só é possível evitar que os estudantes de Medicina pratiquem os seus conhecimentos recém-adquiridos em pacientes reais (tornando, na maior parte das vezes, as intervenções mais demoradas e dolorosas do que as habituais com cirurgiões experientes) como também passa a ser exequível o apoio a uma intervenção cirúrgica na altura e no local em que esta ocorre.

Este é o caso da cirurgia ao fígado realizada por uma equipa médica da Alemanha, recorrendo à RA para visualizar informações sobre o planeamento da mesma, nomeadamente identificando algumas estruturas críticas como vasos e tumores [17].

Na Universidade Tecnológica de Munique, por sua vez, estão a ser desenvolvidos estudos sobre a utilização da RA em cirurgias em que o médico usa um HMD para operar os pacientes com a menor incisão possível e com maior precisão, já que o HMD permite a exibição de um modelo tridimensional virtual do órgão a ser operado na posição real do órgão do paciente [18].

Investigadores da Universidade de Washington desenvolveram uns óculos de RA que permitem detetar as células cancerígenas existentes num tecido, projetando a informação da sua localização e concentração no campo de visão do cirurgião. Esta tecnologia já foi usada em cirurgias de cancro da pele e da mama. Para além de permitir a sua identificação e suportar a sua extração, esta aplicação evita a remoção das células saudáveis que se encontram na proximidade do tumor, uma vez que permite que o cirurgião as distinga das células afetadas [19].

Os Google Glass, lançados ao público em 2014, vieram alargar as possibilidades de uso da RA em pleno bloco operatório, uma vez que permitem a adição de informação importante sobre o paciente durante a operação, sem ocupar as mãos do cirurgião e sem necessidade de interromper a operação. A informação dada tanto pode ser uma indicação sobre o estado de um órgão do paciente como uma representação dos seus batimentos cardíacos, da sua pressão arterial ou de um detalhe médico que pode influenciar a

administração de alguma medicação, como uma doença genética ou uma alergia. Na Figura 3.5 é possível visualizar algumas das indicações que poderão ser disponibilizadas pelos Google Glass a nível cirúrgico. Estas informações foram geradas para uma prova de conceito da Philips Healthcare [20][21][22].



Figura 3.5 - Google Glass aplicados à Medicina [21][22]

Também com os Google Glass, investigadores da Universidade da Carolina do Norte procuraram uma forma de exibir o resultado de uma ultrassonografia sobre o corpo do próprio paciente. Esta necessidade adveio do facto dos médicos estarem normalmente obrigados a interpretar as imagens 2D exibidas por uma ultrassonografia numa tela. Como tal, não só precisam de desviar os olhos do paciente enquanto as analisam e, baseados nas suas experiências médicas, identificarem possíveis problemas, como necessitam de “integrar mentalmente” as imagens exibidas na tela com o corpo do paciente [23].

Quanto mais pormenorizados forem os modelos tridimensionais e mais forem as possibilidades de interação com os mesmos, melhores serão os seus resultados no apoio à Medicina, já que a proximidade do modelo virtual ao real é fundamental para que os médicos reconheçam as diferentes estruturas num paciente e que os modelos sirvam como guias para a interação dos médicos com os pacientes. A Figura 3.6 mostra algumas das imagens médicas realistas que já se conseguem obter na atualidade, a partir de dados provenientes de ressonâncias magnéticas ou de tomografias computadorizadas [24].



Figura 3.6 - Imagens médicas realistas mostrando estruturas internas do corpo [24]

Ferrari et al. desenvolveram um dispositivo de visualização 3D barato, leve e simples baseado em Realidade Mista que permite a médicos ver exames radiológicos pré-operativos de uma forma natural [25]. Para tal, é usado um HMD equipado com duas câmaras de alta definição (Figura 3.7).

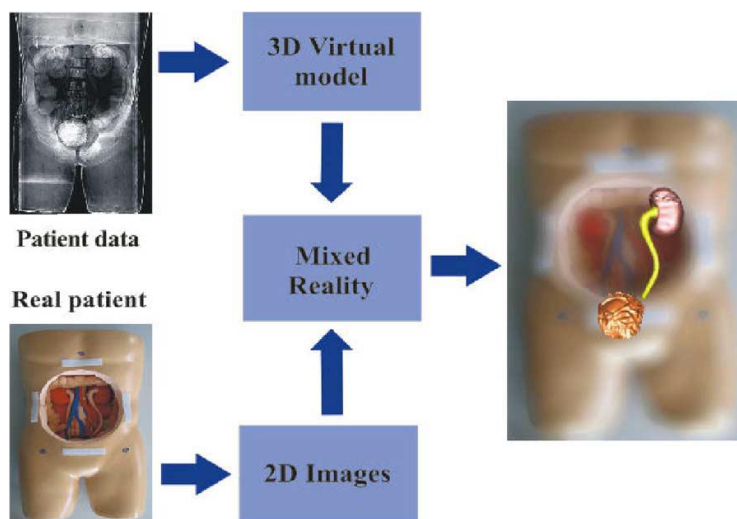


Figura 3.7 – Esquema funcional do sistema de realidade mista proposto por Ferrari et al.[25]

Konishi et al. desenvolveram um sistema de navegação de RA usando um sistema de sensores tridimensional que pode ser utilizado em cirurgia endoscópica. Informação anatômica é obtida e visualizada em tempo real graças a aparelhos ultrassonográficos, juntando-se a imagens de tomografia computadorizada [26].

No que respeita à Neurologia, já foi criado um protótipo de RA, em França, denominado *Mind Mirror* (Figura 3.8) que permite visualizar a atividade cerebral em tempo real. Para tal, sobrepõe imagens da atividade cerebral ao rosto do paciente, antecipadamente equipado de um capacete de elétrodos que permite captar a atividade elétrica, transcrevendo-a sob a forma de imagens [27].



Figura 3.8 - Aplicação *Mind Mirror*, para visualizar em tempo real a atividade cerebral [27]

Os neurocirurgiões estão constantemente a requisitar simuladores que os possam guiar pelos procedimentos, simuladores esses que têm levado a avanços no tratamento de doenças neurológicas. Echegaray et al. desenvolvem um simulador para perfuração do crânio e interação cirúrgica com o cérebro, combinando imagiologia médica, segmentação, *volume rendering* em tempo real, modelação e simulação física, tratamento de colisões, visualização e *feedback* háptico [28]. O simulador, para além de ser usado para treino, permite planear e ensaiar procedimentos neurocirúrgicos complexos.

Cutolo et al. validam uma solução de RA no âmbito da Medicina que usa HMDs e *tracking* baseado em marcas [29]. Obtendo bons resultados de precisão e robustez, conseguiram levar a cabo estudos clínicos sobre a usabilidade e ergonomia da aplicação em cirurgias maxilofaciais e ortopédicas.

Um dos grandes obstáculos para a aplicação clínica da RA é a interação homem-máquina. A visualização de informação desnecessária, obsoleta ou redundante pode confundir o distrair o utilizador, reduzindo a utilidade e a vontade de adotar o sistema. Katic et al. apresentam avanços nesta área, pois propõem um sistema de RA inteligente, capaz de automaticamente filtrar a informação disponível com base em fases reconhecidas da operação [30]. O sistema foi implementado para uso em cirurgia laparoscópica do fígado e da vesícula biliar e alguns elementos podem ser observados na Figura 3.9.

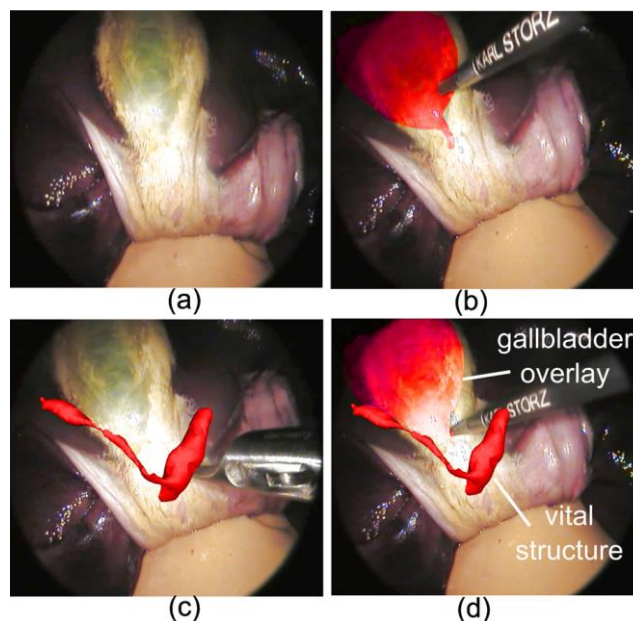


Figura 3.9 - (a) Vista real do fígado, (b) visualização da vesícula biliar, (c) visualização de estruturas de risco, (d) visualização de ambas as componentes [30]

Apesar dos inúmeros estudos e provas de conceito já realizadas, a adoção real de técnicas de RA na Medicina tem sido lenta, principalmente devido a limitações nos equipamentos disponíveis, a insuficiente precisão do *tracking* e alinhamento, à necessidade de algoritmos complexos e a problemas de visualização da informação e nos *displays* usados. Mais que isso, para um sistema de RA ser bem-sucedido é necessário enquadrar-se sem problemas no bloco operatório, providenciar transferência de dados suave e não interferir com o fluxo de trabalho tradicional. Além de tudo isto, é preciso convencer-se os cirurgiões das vantagens reais da inclusão destas tecnologias, tais como tempos de intervenção reduzidos, menor risco para os pacientes, gestão de dados simplificada e, acima de tudo, melhores resultados para o paciente [31].

3.4 Conclusão

Neste capítulo introduziram-se conceitos importantes para o trabalho e apresentou-se um estado-da-arte relativo ao uso da RA em Medicina, que evidencia o potencial desta abordagem. Os capítulos seguintes são relativos à solução apresentada nesta tese.

Capítulo 4

Análise de Requisitos e Trabalho Inicial

Neste capítulo apresenta-se a análise de requisitos do projeto, que inclui a definição dos *Stakeholders*, e dos requisitos funcionais e não-funcionais da solução a concretizar. Descreve-se ainda o trabalho inicial de preparação ao desenvolvimento da aplicação, que inclui a pesquisa efetuada, o estudo sobre o *hardware* e o *software* adequados e a fase de familiarização com o *software* escolhido.

4.1 Análise de Requisitos

Esta secção apresenta a definição dos *stakeholders* do projeto e os requisitos funcionais e os requisitos não-funcionais da solução.

4.1.1 Definição dos *Stakeholders*

Sumário dos *stakeholders* não-utilizadores:

- Ana Paula Cláudio e Beatriz Carmo, envolvidas no desenvolvimento do projeto e orientadoras do mesmo;
- Hugo Ferreira, Médico e Físico, acompanhou o desenvolvimento do projeto, dando orientações relativas à componente médica do mesmo;
- José Soeiro, principal envolvido no desenvolvimento do projeto e aluno de PEI (Projeto de Engenharia Informática).

Sumário dos *stakeholders* utilizadores:

- Médico, utiliza a aplicação como guia na aplicação da técnica de Estimulação Magnética Transcraniana ou como ferramenta de observação/estudo/interação com o modelo virtual tridimensional do cérebro;
- Estudantes de Medicina ou outros interessados na visualização do cérebro.

Objetivos chave dos *stakeholders* utilizadores:

- Médico:
 - Visualizar o modelo virtual do cérebro do paciente e a área a estimular, alinhado sobre a imagem da cabeça do mesmo;
 - Com o auxílio da visualização efetuada aplicar a técnica de EMT sobre o paciente;
 - No modo de Realidade Virtual, observar o modelo virtual do cérebro e interagir com o mesmo;
 - Controlar quais as áreas do cérebro visíveis na aplicação.
- Estudantes de Medicina ou outros:
 - Utilizar a aplicação para aprender mais sobre a estrutura e organização do cérebro.

4.1.2 Requisitos Funcionais

No final do desenvolvimento do projeto, a aplicação fornece as seguintes funcionalidades:

- Modo de Realidade Aumentada
 - Visualização do modelo 3D do cérebro alinhado com imagem da cabeça do paciente, de acordo com os ficheiros introduzidos na aplicação (contendo dados pré-adquiridos, por exemplo, por ressonância magnética) e com o posicionamento correto de uma touca com marcas fiduciais colocada na cabeça do paciente;
 - Calibração das marcas na cabeça do paciente, com o auxílio de indicadores textuais coloridos (este é o procedimento inicial e é necessário aos cálculos de alinhamento do modelo virtual do cérebro com a imagem real);
 - Visualização de uma representação virtual do aparelho de estimulação (um glifo), na posição real do mesmo, de acordo com o posicionamento de uma marca no objeto real;
 - Mudança de cor do glifo do aparelho de estimulação em função da distância a que está da cabeça do paciente.

- Modo de Realidade Virtual
 - Aumento e diminuição de escala do modelo do cérebro, através de um gesto de toque no ecrã do *smartphone*;
 - Rotação segundo um eixo principal do modelo do cérebro, através de um gesto de toque no ecrã do *smartphone*;
 - Alteração do eixo de rotação do modelo do cérebro, entre o eixo horizontal e o eixo vertical do ecrã, através de seleção em botões na interface de utilizador;
 - Posicionamento automático do modelo virtual em vistas predeterminadas (frente, trás, esquerda, direita, cima, baixo), através de seleção em botões na interface de utilizador.
- Menu de Opções
 - Escolha entre um de dois glifos diferentes para representar o modelo virtual do aparelho de estimulação;
 - Seleção de quais hemisférios/lobos/áreas individuais do cérebro se pretende visualizar nos modos de Realidade Aumentada e de Realidade Virtual.

4.1.3 Requisitos Não-funcionais

De seguida descrevem-se os requisitos não-funcionais do sistema:

- Usabilidade: A aplicação deve ter uma interface de utilizador simples, intuitiva e fácil de operar, de modo a que se consiga aceder de forma rápida a todas as funcionalidades da aplicação.
- Desempenho: A aplicação deve funcionar sem tempos de espera observáveis, sendo os modelos virtuais atualizados rapidamente em cada *frame*, para que a visualização seja fluida, especialmente no modo de Realidade Aumentada, e a resposta aos comandos dados pelo utilizador deve ser rápida.
- Restrições de Execução: A aplicação deve ser instalada em dispositivos móveis com o sistema operativo Android.

- Restrições de Desenvolvimento: A aplicação deve ser desenvolvida em Java, com a estrutura de um projeto Android e usando a API deste sistema operativo.

4.2 Trabalho Inicial

4.2.1 Pesquisa efetuada

Numa fase prévia ao início deste projeto, quando a ideia surgiu, foi desde logo iniciado um trabalho de pesquisa e análise para recolher o estado-da-arte das aplicações de RA na Medicina (apresentado no Capítulo 3) e perceber que *hardware* e *software* se deveria usar para desenvolver a aplicação.

4.2.2 Escolha do equipamento

Inicialmente ponderou-se o uso de um *tablet* para o usufruto da aplicação. O ecrã seria maior e mais confortável para a visão do médico. No entanto, forçava o médico a segurar o aparelho com as duas mãos para que a utilização se tornasse mais confortável, o que impedia a aplicação da técnica de estimulação pelo médico. Para além disto, um *tablet* é um aparelho menos portátil que um *smartphone*, sendo que este último é facilmente transportado no bolso, muito menos pesado para uso e é muito mais provável os médicos já possuírem um seu, que levam sempre consigo e onde podem ter a aplicação instalada.

O sistema operativo do equipamento era outra escolha essencial a realizar. Era necessário ter em conta a acessibilidade dos dispositivos desse sistema operativo para os médicos, bem como a conveniência de se desenvolver uma aplicação para esse sistema operativo.

Devido à sua pouca prevalência no mercado de *smartphones*, sistemas operativos como o Windows Phone e o Blackberry foram excluídos e passou-se a considerar apenas os sistemas Android e iOS. Fazendo a comparação, concluiu-se que criar a aplicação para o iOS seria muito mais restritivo, pois todos os médicos teriam de possuir um iPhone para

usar a aplicação. Além disso, criar uma aplicação para o sistema operativo da Apple exigia o pagamento de uma subscrição.

O sistema Android, por sua vez, é usado na grande maioria dos dispositivos móveis, sendo bastante acessível. A criação da aplicação também poderia não envolver nenhum pagamento à Google, pois a Faculdade já possuía uma conta de programador que podia ser usada para publicar a aplicação, e era fácil testar a aplicação num *smartphone*. Para culminar, o desenvolvimento de aplicações Android é feito em Java, linguagem que já se conhecia, e existe também muito suporte *online*, oficial e não oficial, sobre a API (*Application Programming Interface*) Android.

Tendo em conta tudo isto, tomou-se a decisão final de criar a aplicação para o sistema operativo Android. Faltava então escolher que *smartphone* Android adquirir para testar a aplicação durante o projeto.

Fez-se uma análise de mercado em termos de poder de computação, quantidade de memória RAM, tamanho do ecrã, qualidade da câmara traseira, autonomia da bateria e, claro, o preço dos dispositivos. Desta análise, resultou a escolha de um Samsung Galaxy S4 como dispositivo de teste [32].

Paralelamente a esta análise, fez-se a escolha de um suporte ocular para *smartphone* a usar com a aplicação. Como já tinha sido referido, pretendia-se que o médico utilizasse a aplicação com as mãos livres, propósito a que estes suportes dão resposta, para além de proporcionarem uma experiência de RA mais imersiva ao utilizador. Após uma análise que incidiu principalmente na compatibilidade com os tamanhos dos *smartphones*, no preço e também na disponibilidade do suporte, decidiu-se adquirir o Durovis Dive [33].

4.2.3 Escolha da ferramenta de RA

Escolhido o *smartphone* e o suporte ocular, faltava apenas analisar que ferramentas podiam ser usadas para criar, de facto, a aplicação de Realidade Aumentada. Na análise foram comparadas duas ferramentas consideradas mais aptas: o Vuforia [34] e o Metaio [5].

O Vuforia descreve-se como uma plataforma para Android e iOS que permite às aplicações reconhecer imagens a partir de uma base de dados de alvos e gerar experiências de RA a partir desse reconhecimento. O Metaio, por sua vez, apresenta-se como uma solução completa de RA, para Android, iOS e Windows, com capacidades avançadas de *tracking*.

No que diz respeito à disponibilidade dos SDK (*Software Development Kits*), o SDK do Vuforia é gratuito e de fácil instalação, funcionando em conjunto com o ambiente integrado de desenvolvimento Eclipse e as *Android Developer Tools* (ADT) para que se crie uma aplicação Android de RA com todas as ferramentas à nossa disposição. O SDK do Metaio, por sua vez, também funciona com o Eclipse e as ADT e também é gratuito, mas as aplicações criadas vêm com uma marca de água, indicando que a aplicação foi feita com o apoio dessa ferramenta. Para remover a marca de água é necessário comprar uma das licenças, o que acarreta custos inoportáveis para este projeto.

Feita esta análise, o Vuforia permite criar aplicações sem custo de licença do *software* e sem marca de água na aplicação final, mas não se apresenta como solução suficientemente robusta para cumprir o objetivo deste projeto. O Metaio, por sua vez, aparenta ter todos os recursos para criar esta aplicação e pode ser utilizado também sem custos, apenas com uma marca de água na aplicação. Por tudo isto, optou-se por desenvolver a aplicação com o auxílio do Metaio SDK.

4.2.4 Familiarização com o *software* escolhido

Escolhidos os equipamentos e ferramentas a utilizar, realizaram-se as primeiras experiências com o *smartphone* de teste e o Metaio SDK. Essas experiências foram acompanhadas pela leitura com mais atenção dos pontos relevantes da documentação do Metaio SDK e das *Android Developer Tools* (ADT).

Começou-se por se estudar alguns tutoriais sobre aplicações Android no geral, para se ganhar uma melhor noção da sua estrutura e saber preparar e usar o ambiente de desenvolvimento. Como mencionado na secção anterior, o ambiente integrado de desenvolvimento escolhido para o desenvolvimento do código desta aplicação foi o Eclipse, juntamente com as ADT.

O código da aplicação está organizado num projeto Android, com todas as suas componentes, o que inclui ficheiros XML, classes Java e outros recursos, como imagens. O Metaio SDK é disponibilizado na forma de um outro projeto Android que se importou para o Eclipse e que funciona como biblioteca, tendo sido ligado ao projeto da aplicação para que todas as classes e métodos que vêm definidos no SDK pudessem ser usados pela mesma.

O *download* do Metaio SDK traz ainda consigo um outro projeto Android, correspondente a uma aplicação “SDKExample” com vários exemplos de utilizações do SDK, isto é, várias demonstrações de Realidade Aumentada. Sendo importado para o Eclipse, é possível compilar esse projeto e correr a aplicação no *smartphone* de teste. No *website* do Metaio [5] existem tutoriais explicativos sobre cada um dos exemplos e como foram criados. Foram lidos os tutoriais e analisado o código dos exemplos mais relevantes, o que permitiu descobrir e compreender melhor, por exemplo, como criar aplicações estereoscópicas com o SDK.

4.3 Conclusão

Neste capítulo apresentou-se a análise de requisitos do projeto e descreveu-se uma etapa inicial de escolha do equipamento e do *software* adequados à concretização da solução, justificando-se as escolhas efetuadas. O capítulo seguinte é relativo ao desenvolvimento da solução.

Capítulo 5

Aplicação *Brain AR/VR*

A solução concebida e implementada durante o projeto tem o nome de *Brain AR/VR* e proporciona experiências de visualização de modelos virtuais do cérebro em Realidade Aumentada e Realidade Virtual num *smartphone*. A aplicação foi pensada para uso médico, em particular no seu modo de RA que pode guiar um médico durante a aplicação da técnica de Estimulação Magnética Transcraniana. Contudo, a aplicação é útil, por exemplo, a estudantes de Medicina e a qualquer pessoa que tenha interesse em visualizar o cérebro.

Este capítulo começa por descrever a aplicação e a sua arquitetura e debruça-se sobre os principais desafios do desenvolvimento, que incluem: *tracking*, visualização, perceção de profundidade e estereoscopia, aquisição e tratamento dos modelos e introdução prévia da informação na aplicação. Detalham-se ainda aspetos relativos ao funcionamento de cada um dos modos, de RA e de RV, e às funcionalidades implementadas.

5.1 Descrição da Aplicação

A aplicação, que é executada num *smartphone*, fornece dois modos de uso: RA e RV. Na interface de entrada da aplicação (ver Figura 5.1), o utilizador pode escolher entre os dois modos carregando em dois botões (“AR” e “VR”, respetivamente). Existe ainda um terceiro botão (“Options”) para entrar num menu de opções, que pode ser acedido a partir de ambos os modos. Os três botões apresentam-se no canto inferior direito da aplicação e, para não ocupar espaço de visualização, desaparecem após alguns segundos sem o utilizador tocar no ecrã, voltando a aparecer depois de se tocar em qualquer ponto deste. Existe ainda um botão (“x”) para sair da aplicação, no canto superior direito, que se mantém sempre visível na interface.

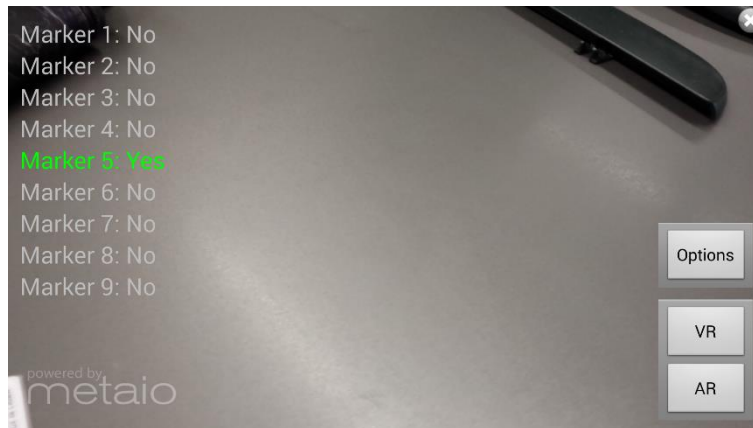


Figura 5.1 - Interface inicial da aplicação, no modo de RA e com os botões de escolha de modo e para o menu de opções visíveis

O modo de RA permite ao utilizador visualizar um modelo virtual 3D do cérebro alinhado sobre a imagem da cabeça de um indivíduo, captada pela câmara do *smartphone*. O alinhamento é feito através do reconhecimento de marcas numa touca colocada na cabeça deste. Idealmente, e para o contexto da EMT, o modelo representará o cérebro do próprio indivíduo, tendo sido obtido a partir de imagens de ressonância magnética, trabalho prévio que tem de ser realizado pelo utilizador médico. As marcas necessitam de um processo de calibração, sendo que uma delas já é calibrada previamente a partir de um ficheiro que se deve guardar no dispositivo móvel. As restantes marcas, devem ser calibradas ao início do uso do modo de RA, existindo indicadores visuais para que o utilizador saiba quais as marcas que ainda lhe falta calibrar (ver Figura 5.2). O processo de calibração encontra-se detalhado na secção 5.3.6.



Figura 5.2 - Indicadores de calibração

A aplicação permite ainda no modo de RA a representação do aparelho de estimulação utilizado na EMT, desde que seja colocada uma marca específica também

sobre este aparelho. Essa representação é um glifo, uma simplificação do aspeto real do aparelho de estimulação e serve para guiar o utilizador e melhorar a percepção de profundidade. O glifo tem ainda um código de cores associado, sendo vermelho quando o aparelho de estimulação está ainda longe do cérebro, amarelo quando a distância é intermédia e verde quando próximo o suficiente para se aplicar a estimulação. Existem duas representações alternativas para o glifo (ver Figura 5.3), que o utilizador pode escolher no menu de opções.

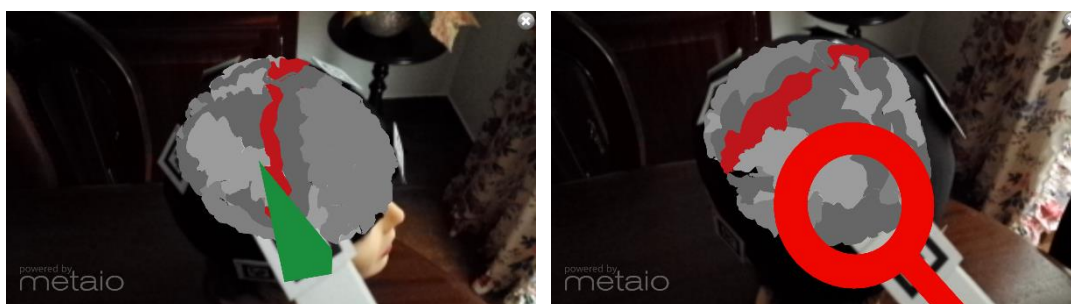


Figura 5.3 - Glifo do aparelho de estimulação nas duas representações: Arrow (esquerda) e Coil (direita)

O modo de RV permite ao utilizador visualizar o modelo virtual do cérebro, observá-lo de vistas predefinidas, rodá-lo e efetuar operações de *zoom-in/out* interactivamente. Neste modo, não se vê a imagem da câmara do dispositivo e a interação é feita por toque, estando dois gestos definidos:

- Colocar dois dedos no ecrã e afastá-los/aproximá-los para, respetivamente, efetuar *zoom-in/out* do modelo;
- Colocar dois dedos no ecrã e movê-los num gesto circular para rodar o modelo num sentido ou noutro, de acordo com um eixo de rotação determinado.

O eixo de rotação pode ser alterado pelo utilizador através de dois botões (“Horiz.” e “Vertic.”) colocados na zona inferior central da interface. À esquerda existem seis outros botões que permitem selecionar vistas predefinidas, isto é, rodar o modelo automaticamente para se adequar a essas vistas: frente (“Front”), trás (“Back”), esquerda (“Left”), direita (“Right”), cima (“Top”) e baixo (“Bottom”). Estes dois grupos de botões são exclusivos do modo de RV e também desaparecem após alguns segundos sem toques no ecrã, reaparecendo quando se toca neste (ver Figura 5.4).

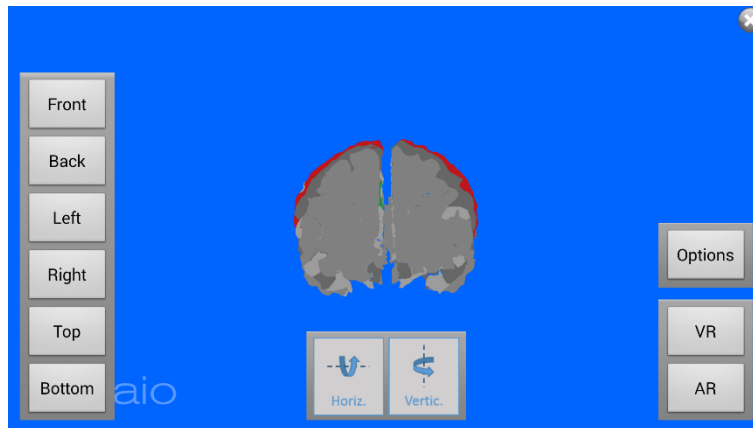


Figura 5.4 - Ecrã do modo de RV, com todos os botões da interface visíveis

A aplicação oferece ainda uma funcionalidade de Visualização Seletiva, disponível no modo de RA e no de RV, que permite ao utilizador escolher as áreas do cérebro que pretende visualizar, ao invés de visualizar o cérebro completo. A escolha das áreas é feita no menu de opções, com uma caixa de verificação para cada uma. O utilizador pode marcar/desmarcar cada caixa para visualizar/ocultar, respetivamente, cada área. Existem caixas de verificação relativas aos dois hemisférios e aos quatro lobos cerebrais, para que se marquem/desmarquem mais rapidamente todas as áreas pertencentes a essas zonas. Na Figura 5.5 pode observar-se o ecrã do menu de opções.

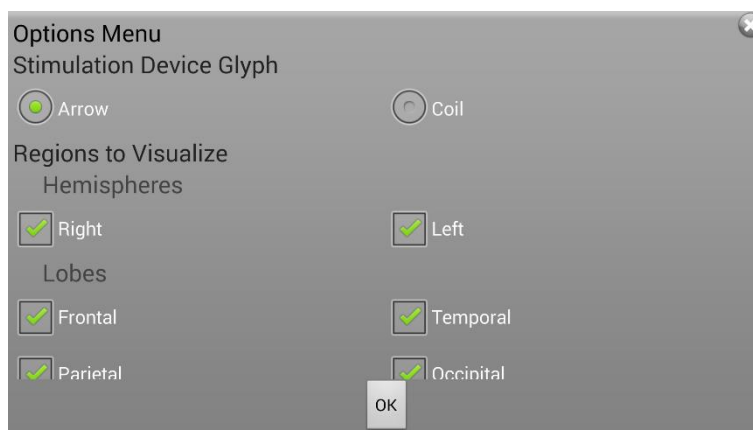


Figura 5.5 - Ecrã do menu de opções

No Anexo A pode encontrar-se o Manual de Utilizador da aplicação, que explica com mais detalhe o seu funcionamento e o trabalho prévio necessário à sua utilização.

5.2 Arquitetura da Solução

A solução implementada tem várias componentes que se interligam. Uma componente lê os toques do utilizador nos botões e desencadeia as respostas adequadas, quer seja uma mudança entre os dois modos (RA e RV), uma entrada no menu de opções ou a visualização do modelo do cérebro a partir de uma vista predefinida no modo de RV, por exemplo. Os dois modos e o menu de opções são componentes individuais que respondem às interações do utilizador e, no caso do modo de RA, às marcas detetadas na imagem da câmara do dispositivo, produzindo mudanças na visualização, quer em termos da interface, quer dos modelos virtuais.

Uma componente muito importante é o carregador de conteúdos, que procura e lê na memória do telemóvel os ficheiros necessários para que a aplicação funcione, informação essa que tem de ser preparada previamente. Os modelos virtuais do cérebro de cada paciente têm de ser obtidos e tratados e só depois guardados no telemóvel a usar para correr a aplicação. Para possibilitar a deteção e visualização distinta das diferentes áreas e também a funcionalidade de Visualização Seletiva, é necessário que seja criado e guardado também no *smartphone* um ficheiro de texto com certas informações. Ao mesmo tempo, de modo a poder usar o modo de Realidade Aumentada, é necessário preparar o *tracking* que a aplicação vai efetuar, sendo necessário colocar as marcas numa touca na cabeça do paciente antes da utilização da aplicação e atualizar um ficheiro de configuração no que diz respeito ao número de marcas, ao seu tamanho e ainda indicar um deslocamento (de posição e rotação) relativo ao referencial cartesiano de uma marca à escolha para aplicar ao referencial cartesiano onde os modelos virtuais vão ser desenhados. Feita toda esta preparação, a aplicação encarregar-se-á de reconhecer as marcas visíveis através da câmara do telemóvel e usá-las para criar a experiência de RA.

A Figura 5.6 mostra esquematicamente a arquitetura interna da aplicação. No Anexo B está disponível um esquema mais completo, que junta à arquitetura interna o fluxo da informação que é preciso preparar previamente. As várias componentes da aplicação serão descritas nas secções que se seguem.

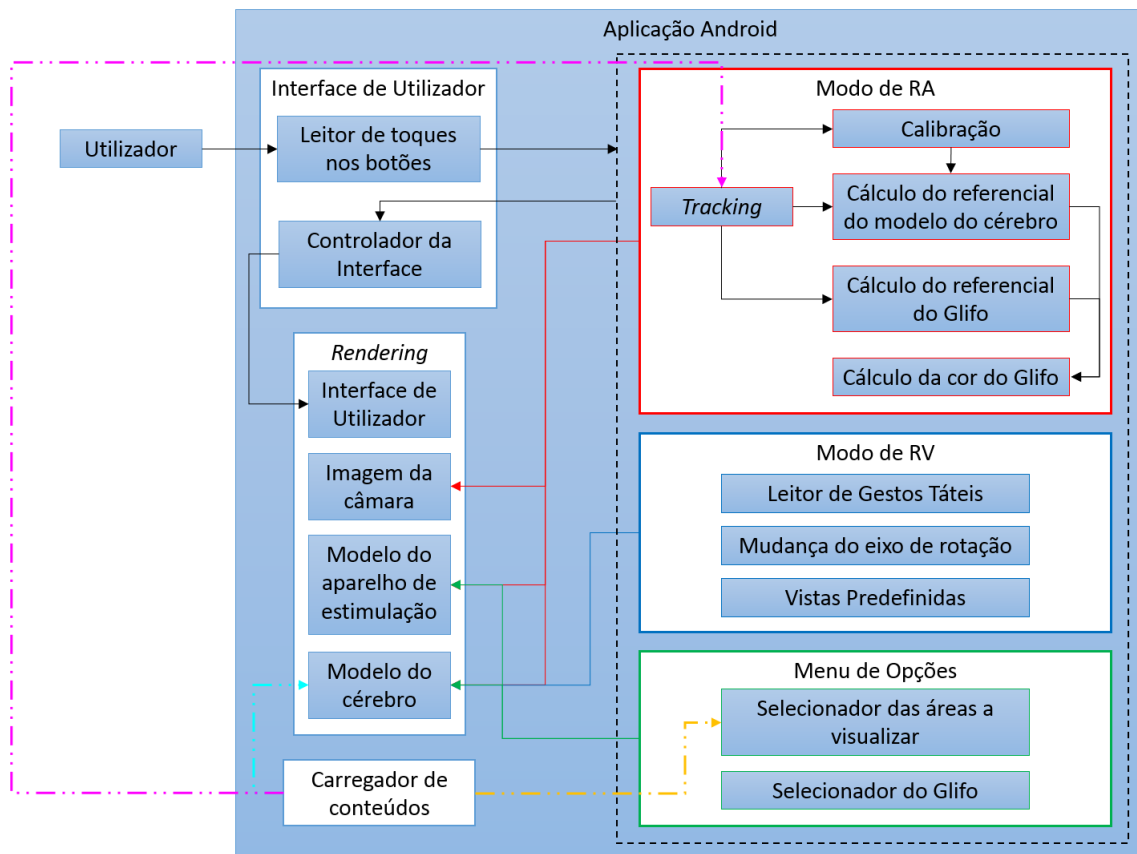


Figura 5.6 - Arquitetura da aplicação

5.3 Tracking

Uma das decisões fundamentais para o bom resultado do projeto era a forma como a aplicação ia fazer o *tracking* – ou seja, o rastreio – da cabeça do paciente. A ideia inicial foi colocar na cabeça do paciente uma touca com códigos QR ou marcas semelhantes, que conseguem ser identificadas por *software* executado em Android. Nesta secção descreve-se:

- a abordagem seguida, consistindo no uso de marcas próprias (secção 5.3.1), apoiado pelo uso de um ficheiro XML de configuração (secção 5.3.2);
- as experiências iniciais efetuadas com as marcas, para a criação do protótipo da aplicação (secção 5.3.3);
- os problemas que esta abordagem acarreta e como foram minimizados (secção 5.3.4);
- o estudo efetuado para se chegar à melhor configuração de marcas na touca (secção 5.3.5);

- o processo de calibração das marcas que foi implementado para resolver alguns problemas que a abordagem inicial tinha (secção 5.3.6);
- o teste efetuado para estudar a influência da iluminância sobre as marcas e da distância às mesmas no *tracking* (secção 5.3.7);
- uma abordagem alternativa ao uso de marcas que se estudou, utilizando os sensores do *smartphone* (secção 5.3.8);

5.3.1 Marcas e seus referenciais

Após algumas experiências e leitura da documentação do Metaio, descobriu-se a existência de marcas próprias fornecidas pela empresa, os *ID Markers* [35], do género de códigos QR, distintas e numeradas de 1 a 512, que o Metaio SDK permite à aplicação reconhecer na imagem da câmara e que podem ser usadas com facilidade para rastrear as várias posições e orientações pretendidas. As marcas, da forma como são disponibilizadas, são quadradas com 60 mm de lado, podendo ser impressas com outro tamanho considerado conveniente. O tamanho das marcas que estão a ser usadas é um fator importante, pois o *software* utiliza essa informação para calcular a distância a que se encontra da marca.

O alinhamento do modelo virtual do cérebro é feito em função da posição e da orientação das marcas, tendo cada uma um referencial cartesiano associado. Em cada instante, o Metaio SDK analisa a imagem captada pela câmara para tentar detetar a existência de marcas. Caso alguma marca seja reconhecida, é usado o referencial da marca como sistema de coordenadas para desenhar o modelo virtual.

Caracterização das marcas:

Na Figura 5.7 pode observar-se uma marca de 60 mm. O ID da marca, neste exemplo a marca número 5, está indicado na margem inferior da mesma e também foi escrito manualmente no canto superior esquerdo da marca, para ficar mais visível, uma vez que é importante que o utilizador consiga identificar cada marca durante o processo de calibração. O ID identifica univocamente a marca, sendo que marcas com IDs diferentes correspondem a marcas com padrões diferentes no centro. Esses padrões de quadrados pretos e brancos, envoltos numa margem preta, são a componente que

possibilita a detecção da marca pela aplicação, tendo de estar completamente visíveis e reconhecíveis na imagem da câmara para que o *tracking* da marca tenha sucesso. O tamanho da marca está indicado junto ao ID, na margem inferior, e indica a medida de lado do padrão central, ou seja, a distância entre duas arestas interiores da margem preta. Essa margem preta e o bordo branco exterior são gerados com tamanho proporcional ao tamanho da marca. Na marca estão também marcados os sentidos positivos dos eixos 'x' e 'y' do referencial da marca. A origem do referencial está no centro do padrão e o eixo 'z' é perpendicular ao plano da marca, tendo o sentido positivo do lado da frente da marca.

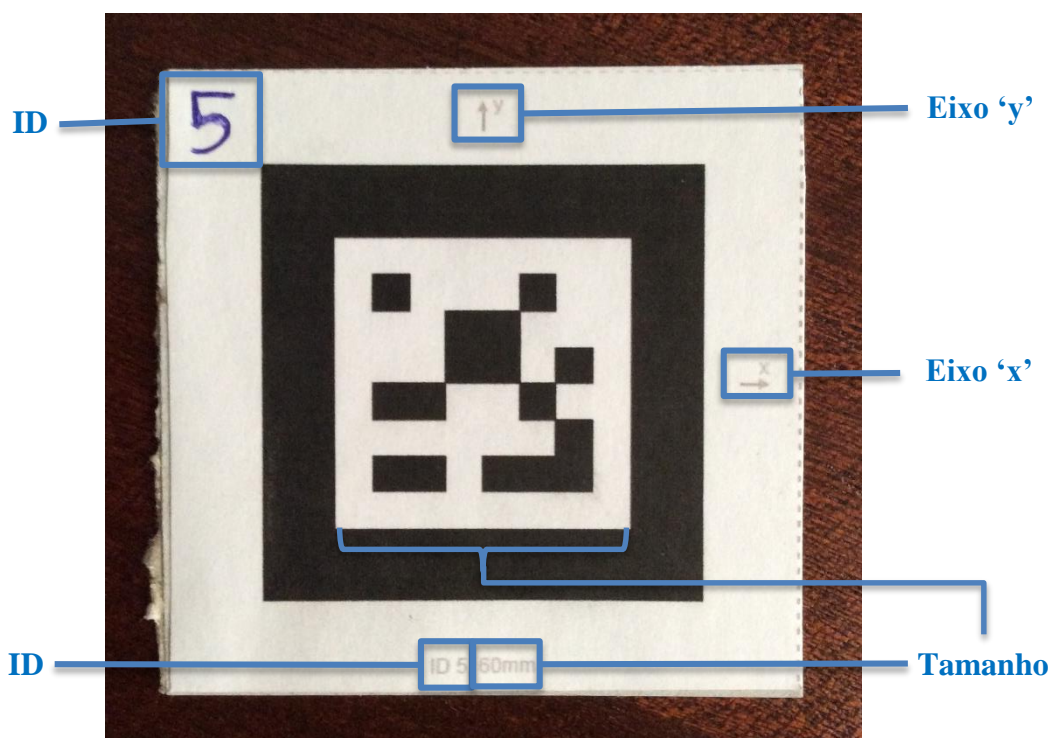


Figura 5.7 – Marca de 60 mm com ID 5 e o sentido dos eixos 'x' e 'y' assinalados

Alinhamento dos referenciais:

Para que o *tracking* se mantenha à medida que o médico se movimenta em torno do paciente, é preciso existirem marcas para serem lidas em vários pontos da cabeça do paciente. Inicialmente, pensou-se que 5 marcas seriam suficientes: quatro ao redor da cabeça (à frente, atrás e dos dois lados) e uma no topo. No entanto, outras configurações foram estudadas para se chegar a uma configuração ideal (ver secção 5.3.5).

É possível atribuir uma transformação (translação e rotação) ao referencial de uma marca, permitindo que, por exemplo, se uma marca estiver colocada sobre a testa do

paciente, o seu referencial esteja centrado não sobre a testa mas num ponto interior à cabeça do mesmo. Esta possibilidade é fundamental para permitir o uso de várias marcas, como pretendido, pois o modelo tem de ser representado sempre na mesma posição e orientação relativa à cabeça do paciente, independentemente da marca que é lida. Assim, ao definir transformações precisas para cada marca, é possível alinhar todos os referenciais para que tenham todos a mesma posição e orientação, como se fossem um único referencial no qual o modelo virtual do cérebro é representado. A definição de quais marcas devem ser detetadas pela aplicação e quais as transformações a atribuir aos referenciais de cada uma, entre outras coisas, é feita num ficheiro de configuração do *tracking*, como descrito na secção 5.3.2.

Para que o alinhamento dos referenciais das várias marcas funcione é fundamental que os valores atribuídos para as transformações dos vários referenciais sejam medidos de forma precisa. A ideia inicial para alcançar esse objetivo era que o paciente tivesse as marcas colocadas – nas mesmas posições e orientações em que seriam usadas posteriormente com a aplicação – durante a ressonância magnética a partir da qual o modelo virtual do cérebro do paciente é obtido. Colocando-se marcas fiduciais – como por exemplo azeite ou outra gordura – sobre as marcas do Metaio, as posições das mesmas ficariam registadas junto ao modelo virtual. Assim, usando *software* de modelação como o Blender para observar o modelo, as relações entre as marcas poderiam ser medidas com precisão, podendo assim ser calculadas as transformações a aplicar ao referencial de cada uma para que todos coincidam num só referencial central, transformações essas que podem depois ser inseridas no ficheiro de configuração do *tracking*.

Colocação das marcas:

Vê-se então que a colocação das marcas no paciente é um passo fundamental para o bom funcionamento do modo de RA. O sucesso do alinhamento do modelo depende de se conseguir manter as marcas nas mesmas posições entre a ressonância magnética e a utilização da aplicação. Assim, pensou-se em colocar as marcas numa touca, que o paciente teria na cabeça em ambas as fases. Isso permite que as posições relativas das marcas não se alterem para o mesmo paciente, sendo que a touca se adapta aos formatos e tamanhos da cabeça de cada paciente. Restaria apenas conseguir-se colocar a touca sempre na mesma posição na cabeça do paciente, de acordo com a numeração das marcas

existentes na touca, o que seria tarefa do médico. É ainda importante referir que a touca não pode, claro, prejudicar em nada os procedimentos da ressonância magnética e da EMT que se pretende fazer.

Ao longo do desenvolvimento do projeto, foi ficando claro que este processo de medição das transformações das marcas seria muito pouco prático, pois implicaria a colocação da touca no paciente durante a ressonância magnética e a obtenção de valores de transformação entre o referencial de cada marca e um referencial comum no interior da cabeça do paciente, valores esses que teriam de ser inseridos no ficheiro de configuração do *tracking*. Assim, uma nova abordagem foi utilizada, em que apenas se define no ficheiro de configuração a transformação do referencial de uma das marcas, que fica calibrada, e as restantes transformações são calculadas e aplicadas em tempo de execução da aplicação, através da comparação da posição e orientação das restantes marcas com a posição e orientação da marca já calibrada no ficheiro de configuração. A este processo dá-se o nome de calibração e é descrito com mais detalhe na secção 5.3.6.

5.3.2 Ficheiro de configuração do *tracking*

O *tracking* a efetuar pela aplicação é configurado num ficheiro XML, lido por um método do Metaio SDK, que tal como o modelo tem de ser guardado no *smartphone*. Este ficheiro contém duas secções principais: <Sensors>, em que é definido o tipo de *tracking* a ser realizado, e <Connections>, onde se definem um ou mais referenciais para serem usados na aplicação, com base no *tracking* que é realizado.

Na secção <Sensors> é definido o elemento <Sensor>, que corresponde ao tipo de *tracking* a efetuar (no nosso caso, "MarkerBasedSensorSource"). Dentro desse elemento <Sensor> definem-se então diferentes elementos <SensorCOS>, correspondentes às marcas a detetar, identificadas a partir do seu ID. É aqui que é inserida também a informação relativa ao tamanho real de cada marca, em milímetros.

Na secção <Connections> são definidos um ou mais elementos <COS>, que correspondem aos referenciais que a aplicação vai usar para representar os modelos virtuais. As posições e orientações destes referenciais são obtidas em cada instante de execução da aplicação a partir das posições e orientações dos referenciais das marcas detetadas, sendo a ligação entre os referenciais das marcas e estes referenciais definida

nesta secção do ficheiro XML. Essa ligação é feita através da associação a cada elemento <COS> de um ou vários elementos <SensorSource>, cada um deles correspondendo a um dos elementos <SensorCOS> definidos acima e associados às marcas.

Assim, pode definir-se um elemento <COS> para cada elemento <SensorSource> – ou seja, um referencial para cada marca –, o que em termos de código da aplicação implica atualizar constantemente o referencial a que o modelo virtual está associado, para que o modelo esteja sempre associado a um referencial cuja marca esteja a ser detetada no momento. Para facilitar, outra hipótese é associarem-se todos os elementos <SensorSource> ao mesmo elemento <COS> – ou seja, todas as marcas ao mesmo referencial –, pelo que em termos de código o modelo virtual pode ficar associado sempre ao mesmo referencial, cuja posição e orientação são calculadas a partir da deteção de qualquer uma das marcas.

Independentemente de qual das abordagens é seguida, a cada elemento <SensorSource> tem de ser aplicada uma transformação para que a posição e orientação do modelo virtual seja constante, independentemente da marca que está a ser detetada pela aplicação. Essa transformação é definida através de um elemento <COSOffset>, onde se podem definir os elementos <TranslationOffset> e <RotationOffset>. De forma simples, a ideia é que se apliquem translações e rotações aos referenciais de cada marca para que todas as marcas passem a apontar para “um mesmo referencial”, sempre na mesma posição relativa às marcas, no qual será desenhado o modelo do cérebro.

Na Figura 5.8 pode observar-se a estrutura geral do ficheiro de configuração e dos elementos <SensorCOS> e <SensorSource>.

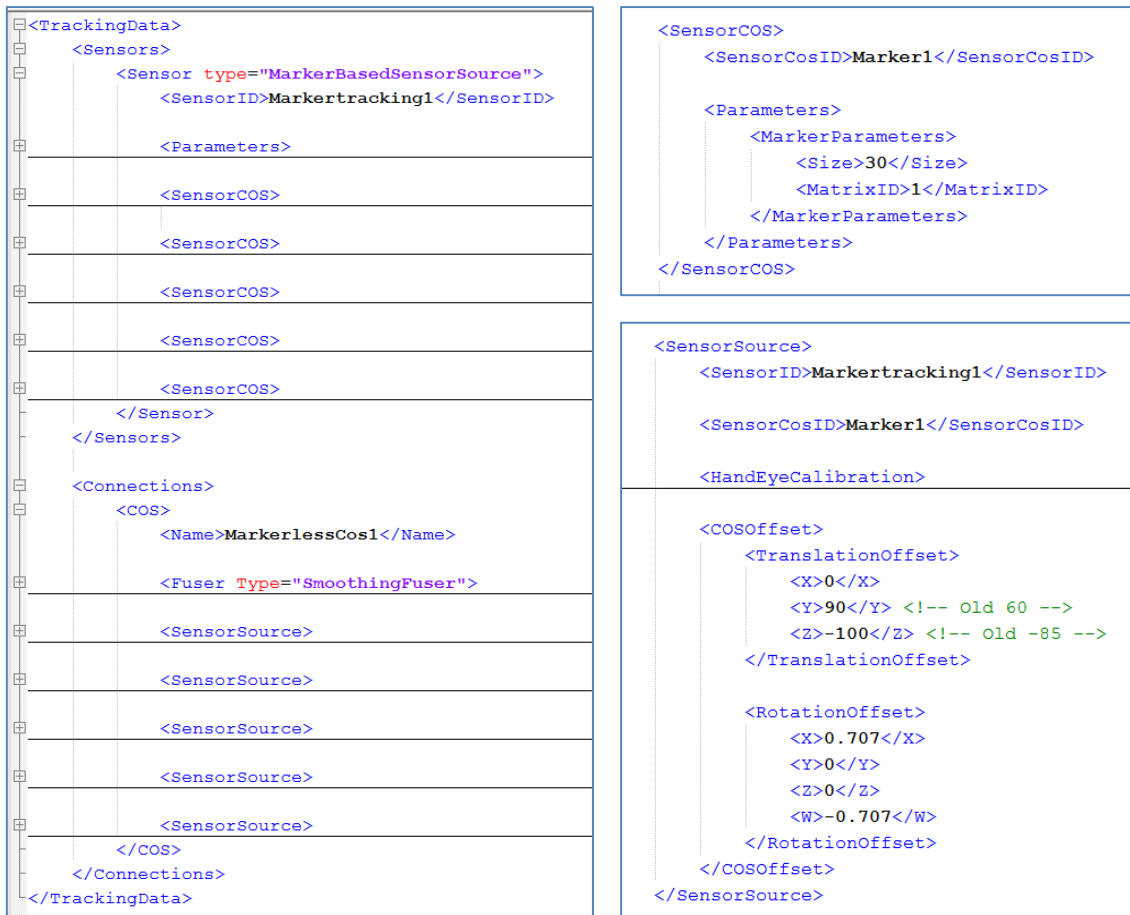


Figura 5.8 – Esquerda: Estrutura global do ficheiro de configuração de tracking com 5 marcas; Em cima à direita: Estrutura de um elemento <SensorCOS>; Em baixo à direita: Estrutura de um elemento <SensorSource>

5.3.3 Protótipo inicial da aplicação

Quando se percebeu a possibilidade de usar os *ID Markers* do Metaio para fazer o *tracking* da cabeça do paciente e se traçou todo o plano de funcionamento da aplicação, conseguiu-se alterar o código e o ficheiro XML de configuração do *tracking* com relativa facilidade. Para definir as rotações dos referenciais das marcas, de modo a ficarem todos com a mesma orientação, foi necessário trabalhar com quaterniões, a forma usada pelo Metaio para representar rotações [36]. Imprimiram-se cinco *ID Markers* (Figura 5.9), numerados de 1 a 5, e a funcionalidade base do *tracking* estava pronta a testar. Faltava apenas ter um “paciente” para se fazer o teste da ideia. Recorreu-se então a um modelo físico de cabeça feminina, com uma touca na qual foram coladas as cinco marcas. Este modelo foi usado para teste durante todo o período de desenvolvimento da aplicação.

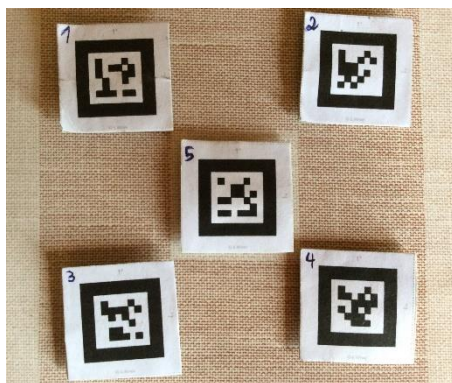


Figura 5.9 – Os cinco primeiros ID Markers fornecidos pelo Metaio

Tiraram-se então medidas de distância de Manhattan [37] entre as várias marcas, para se calcularem as translações a aplicar aos referenciais de todas as marcas para coincidirem num mesmo referencial, que se considerou na altura sobre a marca 5, a do topo da cabeça. Inseriram-se então esses valores no ficheiro XML de configuração do tracking, trasladando e rodando os referenciais das 4 primeiras marcas para coincidirem com o referencial da marca 5. Na Figura 5.10 pode observar-se o modelo físico de cabeça feminina com a touca e as marcas nela colocadas.

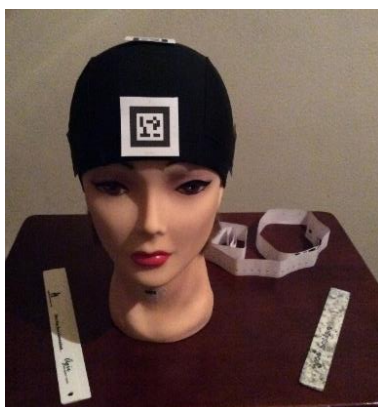


Figura 5.10 - O modelo físico de cabeça feminina durante o processo de medição, com as marcas colocadas na touca

Feito este trabalho, executou-se a aplicação no telemóvel e efetuou-se o teste. Nesta fase, utilizou-se um modelo virtual 3D que vinha com a aplicação exemplo do Metaio SDK, mas mais tarde utilizaram-se modelos do cérebro disponibilizados pelo projeto Brainer [38], como explicado na secção 5.7.2.

Os resultados obtidos pelo tracking na altura não se revelaram perfeitos, por motivos que se discutem nas secções seguintes. Contudo, este protótipo inicial permitiu concluir que a ideia funcionava e serviu de base para o que se fez durante o resto do projeto.

5.3.4 Problemas com o *tracking* de marcas

Após ter um protótipo da aplicação concretizado, verificou-se que o *tracking* efetuado pela aplicação ainda não funcionava perfeitamente, o que levava a que o modelo virtual nem sempre ficasse corretamente alinhado com a cabeça do paciente e outras vezes desaparecesse completamente por instantes, voltando a aparecer em seguida. Estes problemas são inerentes à abordagem seguida, de usar marcas para efetuar o *tracking* da cabeça do paciente. Outras abordagens foram tentadas, como é descrito na secção 5.3.8, mas o uso das marcas acabou por prevalecer como a melhor opção. Ainda assim, analisemos aqui os problemas associados com esta abordagem.

Um dos motivos para o alinhamento do modelo virtual não ter sido perfeito com o protótipo inicial prende-se com o facto de as transformações dos referenciais das marcas, inseridas no ficheiro XML de configuração do *tracking*, terem sido obtidas com medidas tiradas manualmente, medindo-se as distâncias entre as marcas sobre o modelo de cabeça feminina, com réguas. Este método torna difícil a obtenção de medidas exatas, o que faz com que a posição do referencial do modelo virtual varie ligeiramente consoante a marca que é lida pela aplicação, pois os referenciais das marcas não coincidem perfeitamente. Como já referido, este problema seria resolvido ao obter as medidas exatas das relações entre as marcas, a partir dos dados da ressonância magnética, se o paciente já levar as marcas colocadas, juntamente com outras marcas fiduciais, durante esse procedimento. Como essas medições ainda exigiriam algum trabalho ao médico, um sistema de calibração das marcas foi implementado, fazendo com que apenas seja necessário introduzir no ficheiro XML as medidas para uma das marcas, enquanto as outras serão calibradas a partir dessa ao início da aplicação. Este processo é descrito na secção 5.3.6.

Outro dos motivos para estas imperfeições consiste nos movimentos bruscos do *smartphone*, que levam a que o programa não consiga reconhecer as marcas nas imagens da câmara e, conseqüentemente, o modelo virtual desapareça. Assim que a câmara volta a estabilizar a aplicação volta a detetar as marcas e posiciona o modelo corretamente. Para além disso, também a luminosidade do ambiente e a distância às marcas influenciam a capacidade de deteção das marcas por parte do *software*. Estes dois últimos fatores de influência foram estudados mais detalhadamente, trabalho que é descrito na secção 5.3.7.

Outra das razões para o modelo virtual desaparecer é o estado das marcas. Com o uso, as marcas acabam por ficar dobradas, ou mesmo sujas, o que altera as suas propriedades visuais e pode fazer com que a aplicação não as consiga reconhecer. Para resolver este problema, terá de se ter cuidado com a manutenção das marcas e, quando necessário, proceder à sua substituição. Como exemplo, quando se efetuou o protótipo da aplicação colaram-se as marcas em papel diretamente na touca, com fita-cola. Assim, as marcas ficaram bastante suscetíveis a ganharem vincos com o uso, para além de que facilmente ocorrem mudanças ligeiras de posição. Com o tempo, isto acabou por acontecer e o desempenho da aplicação deteriorou-se. Como tal, substituíram-se as marcas por marcas coladas em cartão, o que as tornou mais resistentes à dobragem. Na Figura 5.11 pode observar-se a touca com as cinco marcas em papel e em cartão.



Figura 5.11 - Touca com as marcas só em papel (esquerda) e em papel coladas em cartão (direita)

Para além de todos estes motivos, acontece também que por vezes a posição da câmara não permite que se detete qualquer marca, por nenhuma aparecer na imagem da câmara ou por aparecer mas vista a partir de um ângulo que a torna irreconhecível. Assim, pode ser necessário ter mais marcas colocadas na touca, em pontos diferentes, para melhorar a cobertura e, conseqüentemente, o *tracking*. Acontece, no entanto, que marcas grandes podem dificultar a colocação de muitas mais marcas. Por outro lado, o uso de marcas mais pequenas pode prejudicar as capacidades de deteção por parte do *software* a distâncias em que ainda é útil que as marcas sejam reconhecidas. Face a isto, fez-se um estudo para se chegar à melhor configuração de marcas, isto é, ao tamanho e número ideal de marcas a colocar na touca. Este estudo é descrito na secção seguinte.

5.3.5 Estudo das configurações de marcas

Como vimos, havia interesse em estudar o desempenho do *tracking* em função do número de marcas usadas e do seu tamanho. Como um maior número de marcas obriga a um menor tamanho das mesmas e esta diminuição de tamanho pode degradar o desempenho da aplicação, interessava perceber até que ponto se podia diminuir o tamanho das marcas e aumentar o seu número. Por outro lado, interessava perceber se era, de facto, útil aumentar o número de marcas e até que ponto é que esse aumento era relevante.

Fez-se então um estudo do desempenho do *tracking* em função do número de marcas e do seu tamanho. Em relação ao número de marcas, testaram-se as seguintes configurações:

- 5 marcas – a configuração do protótipo inicial, com marcas à frente, atrás, à esquerda, à direita e no topo da cabeça;
- 9 marcas – foram colocadas quatro marcas adicionais, nas diagonais;
- 15 marcas – foram acrescentadas mais quatro marcas acima das quatro laterais iniciais e mais uma marca de cada lado da marca traseira inicial.

Em relação ao tamanho das marcas, testaram-se marcas dos seguintes tamanhos: 15, 30 e 45 mm de lado. Para as marcas de 15 mm, testaram-se as configurações com 5, 9 e 15 marcas. Para as marcas de 30 mm, testaram-se as configurações com 5 e 9 marcas, mas não a com 15 marcas, pois estas eram demasiado grandes para isso. Para as marcas de 45 mm, também devido ao seu tamanho, apenas se testou a configuração de 5 marcas. Estas marcas de 45 mm são, de facto, demasiado grandes para serem usadas no contexto deste projeto, mas decidiu-se fazer o teste com elas na mesma, para se analisar a influência que marcas de tal tamanho tinham no *tracking* efetuado, conhecimento esse que pode ser útil noutras aplicações.

Para além destes seis casos de teste, analisou-se ainda o caso usado com o protótipo inicial, em que as marcas eram de 20 mm na configuração de 5 marcas. Na Figura 5.12 podem observar-se as diferentes configurações estudadas.

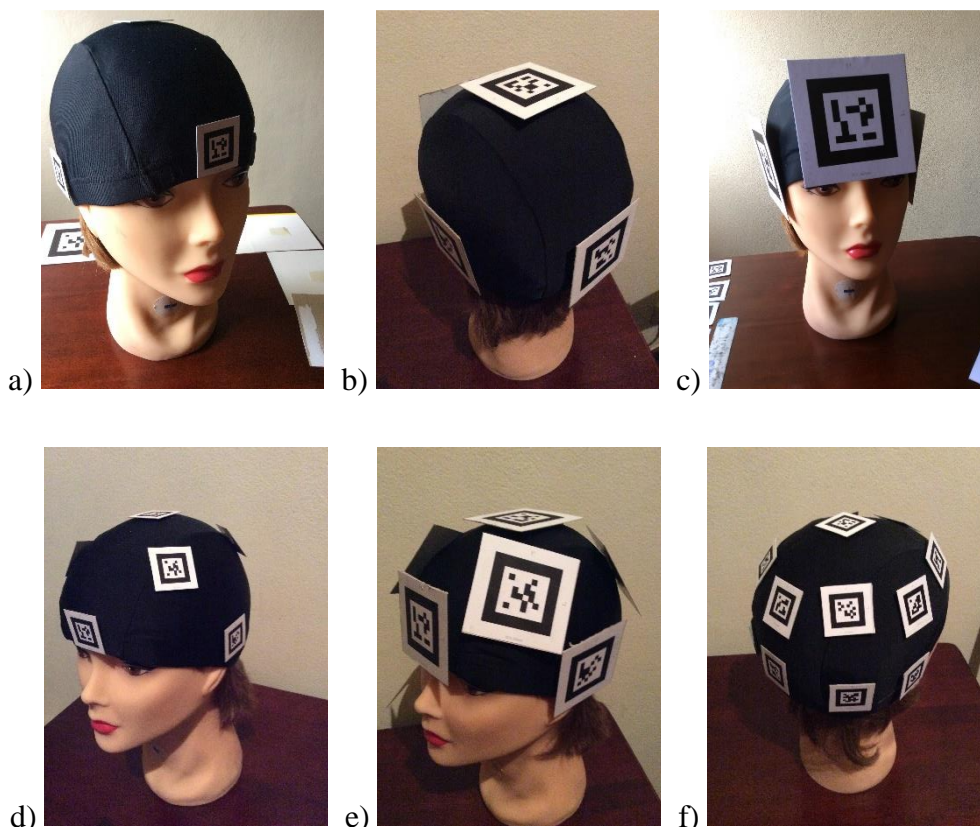


Figura 5.12 - a) 5 marcas de 15 mm; b) 5 marcas de 30 mm; c) 5 marcas de 45 mm; d) 9 marcas de 15 mm; e) 9 marcas de 30 mm; f) 15 marcas de 15 mm

As variáveis medidas em cada teste são o número de *frames-per-second* (FPS) do *tracking*, da câmara e do *rendering*. O número de FPS do *tracking* permite saber quantas vezes por segundo a aplicação deteta as marcas e calcula o centro do referencial e é o valor que mais interessa neste teste. O número de FPS da câmara indica quantas vezes por segundo a imagem da câmara é capturada para ser analisada, enquanto o número de FPS do *rendering* permite saber quantas vezes por segundo os modelos virtuais são desenhados no ecrã.

Na Tabela 1 mostram-se os resultados para os vários casos de teste.

Tabela 1 - Resultados do estudo de configuração das marcas

	<i>Rendering</i> FPS	<i>Tracking</i> FPS	Câmara FPS
5 marcas de 20 mm	53,2 fps	15 fps	14,6 fps
5 marcas de 45 mm	59,6 fps	29,2 fps	29,2 fps
5 marcas de 30 mm	47 fps	14,6 fps	14,4 fps
5 marcas de 15 mm	49,4 fps	16,6 fps	15,4 fps
9 marcas de 30 mm	55,4 fps	15,2 fps	15 fps
9 marcas de 15 mm	49,6 fps	15,6 fps	15,6 fps
15 marcas de 15 mm	47,2 fps	15,4 fps	15,2 fps

Analisando os resultados obtidos, observa-se que o número de FPS do *tracking* varia substancialmente apenas com o tamanho das marcas, duplicando quando se usam as marcas maiores, de 45 mm. De resto, com todos os outros tamanhos, o seu valor mantém-se em cerca de 15 fps, o valor que já se verificava na configuração inicial.

Em relação ao número de FPS da câmara, observa-se uma relação estreita com o número de FPS do *tracking*, estando os dois valores sempre bastante próximos um do outro. Isto explica-se pelo facto de a aplicação retirar uma nova imagem da câmara sempre que acaba de fazer o *tracking* da imagem anterior, daí os dois valores estarem sempre ligados.

Já em relação ao número de FPS do *rendering*, os valores não se alteram muito dos valores usuais na configuração inicial, mas verifica-se um valor maior quando o número de FPS do *tracking* é maior, o que é natural, pois se as marcas são detetadas mais vezes num segundo, mais vezes se inicia o processo de *rendering*. Talvez se possa concluir daí que o desempenho do *rendering* está a ser limitado pelo do *tracking*, mas como a primeira atinge normalmente valores perfeitamente aceitáveis, isto não será razão para sermos forçados a conseguir maiores valores de FPS para o *tracking*.

Dos resultados do teste pode concluir-se, pois, que o tamanho das marcas apenas melhora o desempenho do *tracking* quando se tratam de marcas muito grandes. No entanto, essa melhoria não é necessária, pois o desempenho em termos de FPS de *tracking* é suficiente com os tamanhos mais pequenos. Conclui-se ainda que o número de marcas usadas não influencia o número de FPS de nenhuma das componentes da aplicação. Contudo, isso não significa que o desempenho global da aplicação não seja beneficiado com o uso de mais marcas, simplesmente esse benefício não consegue ser medido com os valores retirados desta experiência.

Na verdade, o uso de mais marcas traz uma grande vantagem. O aparelho de estimulação a ser usado com a aplicação vai estar a ser utilizado sobre a cabeça do paciente, provavelmente tapando algumas das marcas. Usando poucas marcas, principalmente se forem maiores, é mais provável que nenhuma das marcas esteja completamente visível em cada instante, pelo que o modelo do cérebro não será representado. Com mais marcas e de menor tamanho, é mais provável existir sempre em

cada instante uma marca completamente visível pelo *smartphone*, o que é extremamente relevante no desempenho prático da aplicação.

Assim, tendo em conta que o estudo das FPS não indicou uma solução mais adequada e atendendo às considerações empíricas feitas acima, optou-se por privilegiar uma configuração com mais marcas de menor dimensão. Sendo que 15 marcas implicava maior trabalho de calibração (ver secção 5.3.6), optou-se por usar, durante o restante período de desenvolvimento da aplicação, a configuração com 9 marcas de 15 mm.

Neste estudo, as medições feitas do número de FPS do *tracking* acabaram por não ser relevantes para chegar às conclusões pretendidas e recorreu-se então a outras observações mais empíricas. É possível que um outro tipo de medições fosse mostrar resultados mais relevantes e que corroborassem as conclusões tiradas.

5.3.6 Calibração das marcas

Como já visto acima, o alinhamento do modelo virtual do cérebro com a cabeça do paciente depende das transformações aplicadas aos referenciais de cada marca. Estas transformações (consistindo em translações e rotações) têm de ser dadas de modo a que os referenciais de todas as marcas coincidam num único referencial, idealmente num ponto central da cabeça do paciente. Se assim for, quando a aplicação deteta qualquer uma das marcas vai sempre desenhar o modelo virtual na mesma posição relativa à cabeça do paciente, como pretendido.

O grande desafio desta abordagem é a obtenção dos valores para a transformação do referencial de cada marca. Inicialmente, pensou-se em colocar a touca com as marcas no paciente durante a ressonância magnética de onde se vai obter o modelo do cérebro, junto com marcas fiduciais que a ressonância detete. Isto permitia que o modelo virtual guardasse informação que possibilitava fazer posteriormente as medições e inseri-las, para cada marca, no ficheiro de configuração do *tracking*. No entanto, esta abordagem era menos prática de implementar, pois implicava obter uma a uma as medidas para cada marca, para além de que era preciso rigor na colocação da touca para que as marcas ficassem na mesma posição na ressonância magnética e mais tarde durante a execução da EMT. Assim, encontrou-se uma nova solução para simplificar o processo.

Esta solução consistia em introduzir no ficheiro de configuração as medidas de transformação do referencial de apenas uma das marcas na touca, por exemplo, a marca 5, do topo da cabeça. As transformações a aplicar aos referenciais das restantes marcas seriam calculadas em tempo de execução da aplicação, através da comparação entre a posição e orientação dessas marcas e a posição e orientação dos referenciais das marcas cujas transformações já tivessem sido calculadas e aplicadas. A este processo deu-se o nome de calibração e passou-se a considerar como marcas calibradas aquelas para as quais a transformação do referencial já foi calculada (e aplicada) e marcas não calibradas aquelas para as quais esse cálculo ainda não foi feito.

Usando esses termos, a única marca que se encontra calibrada no início da execução da aplicação é a marca do topo, pois a transformação do seu referencial é atribuída diretamente no ficheiro de configuração do *tracking*. Todas as outras marcas encontram-se não calibradas no início da aplicação e antes de poder usar o modo de RA é necessário proceder a essa calibração das marcas. Para isso, o utilizador deve apontar a câmara de modo a que a aplicação consiga detetar ao mesmo tempo uma das marcas não calibradas e uma das marcas já calibradas. Então, a aplicação calibra automaticamente a marca não calibrada a partir da posição da marca que já estava calibrada. A partir daí, a marca recém-calibrada passa a poder ser usada para calibrar as marcas que faltam, até que todas as marcas estejam calibradas. Mais informações sobre os cálculos efetuados para fazer a calibração de cada marca podem ser encontradas no Anexo C.

Para ajudar o utilizador no processo de calibração, foram incorporados na interface da aplicação indicadores textuais coloridos, indicando quais as marcas já calibradas e quais ainda não o estão. Existe uma linha de texto para cada marca, com o número (ID) da marca e a palavra “Yes” ou “No” à frente indicando se a marca está ou não calibrada. Além disso, as linhas de marcas calibradas são representadas a verde, enquanto as restantes ficam a cinzento. Quando todas as marcas ficam calibradas uma mensagem aparece por baixo das restantes linhas de texto, o modelo do cérebro passa a estar visível (se alguma marca estiver a ser detetada) e passados três segundos todos os indicadores auxiliares do processo de calibração desaparecem da interface, para não prejudicar a experiência de visualização. A Figura 5.13 apresenta esta interface.

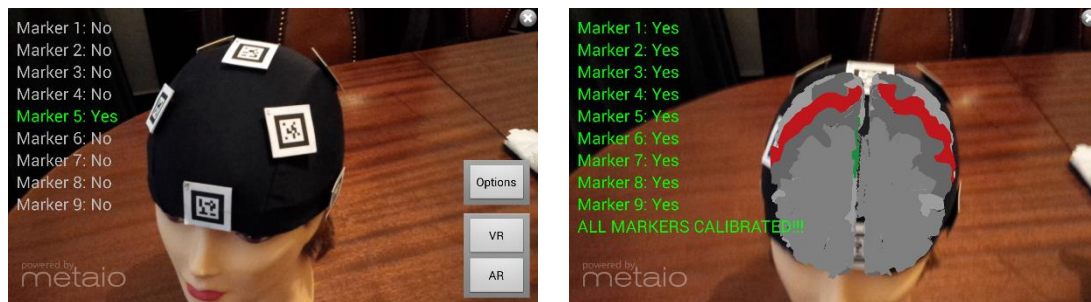


Figura 5.13 – Interface do modo de RA durante o processo de calibração (esquerda) e quando todas as marcas ficam calibradas (direita)

O processo de calibração facilitou muito o trabalho do médico, pois deixou de ser necessário obter medidas de transformação para todas as marcas. Com este processo, o médico apenas precisa de introduzir no ficheiro de configuração do *tracking* a transformação do referencial de uma das marcas, por exemplo a do topo. A medição dessa transformação pode ser efetuada mais facilmente se a marca estiver colocada numa posição de referência que o médico conhece e que sabe localizar no modelo virtual do cérebro do paciente. Neste caso, mesmo sem colocar a touca no paciente durante a ressonância magnética, o médico consegue analisar o modelo virtual tridimensional no seu computador (usando, por exemplo, o Blender) e medir a translação e rotação da marca em relação à origem do referencial em que o modelo foi criado, valores esses que vai introduzir no ficheiro de configuração.

Neste processo, alguma imprecisão nas medidas afeta apenas a definição do referencial em que o modelo virtual do cérebro vai ser desenhado. Os referenciais das restantes marcas, na abordagem anterior, estariam sujeitos igualmente a medições não exatas que fariam com que cada marca apontasse para posições e orientações diferentes. Nesta abordagem, porém, os referenciais das restantes marcas são definidos para apontar todos para o referencial da marca inicial, logo teoricamente o alinhamento do modelo virtual vai ser igual para todas as marcas, mesmo que possa não ser exatamente coincidente com o real.

Infelizmente, as limitações de *tracking* do *software* e das capacidades da câmara do dispositivo provocam erros de aproximação no cálculo das transformações entre os referenciais das marcas, pelo que a calibração também não é absolutamente exata e o alinhamento perfeito de marca para marca também não se verifica. Para tentar minimizar os erros, a calibração de cada marca, que demora uma *frame* a executar, é repetida um número determinado de vezes, calcula-se a média dos valores obtidos em cada *frame* para

as coordenadas da translação e rotação e só depois se utiliza essa média como resultado da calibração da marca. Apesar dos erros não serem completamente eliminados, este processo de calibração automática pela aplicação permite poupar muito trabalho de calibração manual de cada marca pelo médico e obter ainda resultados satisfatórios.

5.3.7 Teste de distância e iluminância

No uso de marcas para efetuar o *tracking* da cabeça do paciente a detecção destas pela câmara do dispositivo pode falhar em alguns instantes, por variadas razões. Uma das mais comuns é uma luz muito forte incidir sobre as marcas, o que impede o seu reconhecimento pela aplicação; outra é a distância da câmara à marca aumentar em demasia, pois a grandes distâncias as marcas ficam pequenas demais na imagem da câmara para o *software* as reconhecer.

Com base nisto, revelou-se pertinente avaliar qual a influência real da intensidade de luz e da distância na capacidade de *tracking* das marcas por parte da aplicação. Para fazer este teste consideraram-se então duas variáveis independentes: a iluminância (medida em lux) e a distância do dispositivo à cabeça do paciente (medida em cm).

A iluminância corresponde ao fluxo luminoso que incide perpendicularmente numa superfície por unidade de área da superfície e é medida com o auxílio de um luxímetro. Na Figura 5.14 podemos ver o luxímetro usado no teste.



Figura 5.14 – Luxímetro usado no teste de distância e luminância

O objetivo deste teste era avaliar a capacidade de detecção de uma marca sob diferentes condições de iluminância e distância. Então, optou-se por se fixar uma das

marcas para a qual a câmara do *smartphone* estava sempre a apontar e medir a percentagem de *frames* da aplicação em que a deteção dessa marca falhava. Para isso foi implementado na aplicação um contador que só era incrementado quando nessa *frame* a deteção da marca falhava, enquanto outro contador era incrementado a cada *frame*. Estas duas variáveis só começavam a contar quando se carregasse num botão que foi adicionado temporariamente à interface e paravam de contar após 500 *frames*. Os valores dos dois contadores eram também apresentados na interface, para serem mais fáceis de registar.

Como a aplicação seria normalmente usada dentro de um edifício, o teste foi também efetuado dentro de portas, numa sala que recebia luz de uma janela. O modelo de cabeça feminina com a touca e as marcas foi colocado em diversos pontos da sala e a diferentes distâncias da janela, de forma a receber diferentes níveis de iluminância. Esses níveis foram medidos colocando-se o luxímetro diretamente sobre a marca, para garantir que se media a iluminância que incidia na marca. Então, posicionado o modelo de cabeça no local correto, usava-se a aplicação com a câmara a apontar para a marca, a partir de uma determinada distância, medida com fita métrica, e iniciavam-se os contadores. Após 500 *frames*, registava-se o número de vezes em que a deteção da marca tinha falhado, como indicado no ecrã, e fazia-se novamente o teste a uma nova distância entre o *smartphone* e a marca. Quando se tinham obtido os valores para todas as distâncias pretendidas, procurava-se uma nova posição para o modelo de cabeça de modo a que sobre a marca incidisse aproximadamente um dos restantes valores de iluminância pretendidos e repetia-se o teste para as várias distâncias.

Para escolher quais os valores de iluminância sob os quais fazer o teste, serviu de referência a discussão de Halsted [39], referida por Gabbard [40], na qual são identificadas gamas de valores de iluminância que correspondem às condições luminosas de diversos ambientes ao ar livre, como se pode ver na Tabela 2.

Tabela 2 - Gamas de iluminância de diversas condições ambiente

Condição ambiente	Iluminância (lux)
Bright sun	50000 – 100000
Hazy day	25000 – 50000
Cloudy bright	10000 – 25000
Cloudy dull	2000 – 10000
Very dull	100 – 2000
Sunset	1 – 100
Full moon	0,01 – 0,1
Starlight	0,001 – 0,001

Iluminâncias de menos de 1 lux correspondiam a condições de muita escuridão em que as marcas não eram visíveis, por isso não se fez o teste com tais valores. Na gama 1 – 100 lux, por se observar que existiam grandes diferenças entre vários valores de iluminância nessa zona, foram considerados para o teste três valores de iluminância nessa gama: 5, 10 e 50 lux. Para as gamas 100 – 2000 lux, 2000 – 10000 lux e 10000 – 25000 lux foi escolhido um valor intermédio para cada uma: 1000, 5000 e 15000 lux. Como o luxímetro disponível apenas detetava valores de iluminância até um máximo de 20000 lux, não foi possível medir valores nas duas gamas com valores superiores a este.

Também interessava estudar condições de iluminância derivadas de luz elétrica, por isso noutra caso de teste fecharam-se todas as fontes de luz natural e testou-se a deteção da marca iluminada apenas por luz elétrica, com uma iluminância de 50 lux.

Em relação às distâncias, fez-se o teste para vários valores entre 5 centímetros e 1 metro. É importante referir que o teste foi efetuado com marcas de 15 mm, aquelas por que se optou para usar com a aplicação, e que com marcas de diferentes dimensões os resultados seriam certamente diferentes. Os resultados do teste podem ver-se na Tabela 3. Em cada célula da tabela encontra-se o número de vezes em que a deteção da marca falhou, nas 500 *frames* do teste, e a respetiva percentagem. Assim, 500 (100%) significa que a deteção falhou em todas as *frames* e 0 (0%) que a marca foi sempre detetada.

Tabela 3 - Resultados do teste de distância/iluminância

Iluminância /Distância	5 lux	10 lux	50 lux (elétr.)	50 lux	1000 lux	5000 lux	15000 lux
5 cm	500 (100%)	454 (90,8%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	326 (65,2%)
10 cm	12 (2,4%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
20 cm	115 (23%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
40 cm	147 (29,4%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
60 cm	325 (65%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)
70 cm	478 (95,6%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	2 (0,4%)	80 (16%)
80 cm	500 (100%)	67 (13,4%)	211 (42,2%)	168 (33,6%)	27 (5,4%)	26 (5,2%)	208 (41,6%)
90 cm	500 (100%)	313 (62,6%)	500 (100%)	327 (65,4%)	161 (32,2%)	196 (39,2%)	275 (55%)
100 cm	500 (100%)	460 (92%)	500 (100%)	500 (100%)	500 (100%)	500 (100%)	500 (100%)

Dos resultados acima várias tendências podem ser observadas. Primeiro, comprova-se que a partir de certa distância a detecção das marcas de 15 mm torna-se bastante difícil, distância essa que é normalmente de cerca de 80 cm, mas que pode diminuir com condições de iluminância demasiado forte ou demasiado fraca. Por outro lado, à distância de 5 cm da marca a capacidade de detecção também piorou, com condições de iluminância demasiado forte ou demasiado fraca. Podendo parecer estranho, este facto justifica-se por a câmara do dispositivo ter dificuldade, nessas condições de iluminância, em manter uma imagem nítida do objeto a tão grande proximidade.

No que diz respeito à iluminância, observa-se que com 5 lux, o que corresponde a um ambiente muito escuro, a detecção da marca é muito fraca, sendo razoável apenas a distâncias próximas de 10 cm mas não a 5 cm, pelo motivo já descrito acima. 10 lux já permitem a detecção da marca para valores intermédios de distância (mais de 5 cm e menos de 80 cm) e entre os 50 e os 5000 lux a detecção funciona bem mesmo para distâncias de 5 cm. Para valores de iluminância de 15000 lux os limites de distância voltam a ficar mais próximos, sendo que distâncias menores que 10 cm e maiores que 70 cm já causam mais falhas na detecção.

Estas conclusões parecem corroborar as observações empíricas de que grandes intensidades de luz a incidir nas marcas dificultam a detecção, no entanto não foi possível medir valores de iluminância suficientemente grandes para que a detecção falhasse nas distâncias intermédias. Isto deveu-se a limitações no luxímetro, que apenas conseguia medir valores até 20000 lux. Contudo, os valores não testados correspondem a condições de iluminação pouco prováveis e facilmente controláveis em ambientes interiores, onde a aplicação será usada.

Em relação ao efeito da luz elétrica, é possível verificar que a detecção da marca piorou a grandes distâncias, comparativamente à detecção da marca com incidência de luz natural com o mesmo valor de iluminância. A diferença não foi muito relevante, porém, e não se verificaram efeitos na detecção da marca a distâncias intermédias.

5.3.8 Teste de uso de sensores

Como já visto acima, o uso de marcas para fazer o *tracking* da cabeça do paciente tem alguns problemas associados e implica um processo de calibração no início da

aplicação, que mesmo assim não resolve todas as imperfeições no alinhamento, devido a erros de aproximação pelo *software*. Assim, algum tempo do projeto foi dedicado à busca de novas soluções que permitissem um melhor *tracking* e alinhamento do modelo.

A grande alternativa ao uso de marcas para fazer o *tracking* numa experiência de RA é recorrer aos sensores do dispositivo móvel para tentar perceber a sua posição ou movimento e atualizar o desenho do modelo virtual no ecrã em função das variações na posição e orientação percebidas pelo *smartphone* relativamente à cabeça do paciente. Usando sensores como o acelerómetro e o giroscópio, as translações e rotações do próprio dispositivo são percebidas e consegue-se então mudar a posição e orientação do modelo virtual no sentido inverso a essas movimentações do dispositivo, para que o modelo aparente ficar sempre no mesmo sítio. É preciso, porém, que a posição inicial do modelo seja obtida por outros meios, sendo o uso de uma marca na cabeça do paciente uma hipótese para que esse alinhamento inicial ocorra.

Com apenas uma marca, não existiriam problemas de calibração e desalinhamento entre os referenciais de várias marcas, no entanto continuaria a ser necessário obter medidas para colocar o referencial dessa marca na posição e orientação corretas. Depois de a aplicação detetar essa marca, os movimentos do telemóvel seriam lidos pelos sensores e usados para atualizar a posição do modelo. Um problema com esta abordagem é que não ajusta o modelo em função de movimentos da cabeça do paciente, pelo que este teria de ficar parado. Uma solução para este problema é a marca de referência ser usada pontualmente pela aplicação para garantir que o modelo virtual está a ser representado na posição e orientação correta no espaço. A frequência com que a marca seria lida poderia ser definida de formas diferentes: ou apenas quando o utilizador o indicasse ou de forma regular segundo um período pré-definido ou mesmo sempre que fosse visível, sendo os sensores usados apenas quando a marca não era detetada.

Esta abordagem de uso de sensores para colmatar as falhas do *tracking* com marcas era interessante, mas faltava perceber se era viável e eficaz, ou seja, se de facto conseguia resolver os problemas e proporcionar uma experiência de RA com menos falhas no alinhamento do modelo virtual.

Foram testados diferentes exemplos disponibilizados pelo Metaio em que o *tracking* é efetuado com o uso de sensores. Estes exemplos, porém, não incluíam o uso

de todos os sensores para calcular o movimento do dispositivo com 6 graus de liberdade, como era necessário: três graus de liberdade para a translação e três graus de liberdade para a rotação; apenas detetavam movimentos mais limitados do dispositivo móvel e logo nesses se viam alguns problemas com o *tracking*, pois a posição do modelo por vezes “saltava” de um ponto para outro no espaço.

Foi ainda testado um outro exemplo de *tracking*, também disponibilizado pelo Metaio, a funcionar com o uso de sensores permitindo movimento com 6 graus de liberdade, como pretendido. Contudo, essa demonstração apenas estava disponível para o sistema operativo iOS. Partindo do código desse exemplo para iOS tentou-se criar um exemplo similar para o sistema operativo Android, para fazer o teste final às capacidades de *tracking* com sensores do *software*. Conseguiu-se fazer esse “*port*” da aplicação exemplo iOS para Android, mas infelizmente os resultados não foram os desejados, pois o exemplo mostrava ainda algumas limitações nas capacidades de *tracking* com precisão.

Após este teste, concluiu-se que o uso de sensores do dispositivo móvel para fazer o *tracking* tinha ainda muitas limitações e não constituía uma alternativa melhor ao uso de marcas. Acima de tudo, as marcas permitem um alinhamento mais seguro e confiável, mesmo que não totalmente exato. Além disso, o uso de sensores tornaria a aplicação dependente dos sensores que estão presentes em cada *smartphone*, para além de que exigiria um grande poder de processamento ao dispositivo. Talvez com mais tempo fosse possível chegar a uma implementação que conseguisse conjugar sensores e marcas para uma experiência melhor de RA, mas pelas razões mencionadas acima e por o calendário do projeto ter prazos apertados foi necessário parar por aqui esta experiência e prosseguir com a abordagem decidida inicialmente.

5.4 Visualização, Perceção de Profundidade e Estereoscopia

Um dos principais objetivos da aplicação *Brain AR/VR* é produzir uma visualização do modelo virtual 3D do cérebro de um paciente sobre a imagem da sua cabeça, num *smartphone*. Esta solução de RA deve guiar o médico na aplicação da técnica de EMT a um paciente. Assim, a visualização feita através do ecrã do *smartphone* é um dos fatores mais decisivos para o sucesso do projeto.

Os modelos virtuais são obtidos *a priori* pelos médicos, por processos de tratamento das imagens de ressonâncias magnéticas feitas ao paciente, como descrito na secção 5.7.1. Esses modelos podem ser tratados usando *software* de modelação 3D como o Blender, onde se podem definir algumas características da visualização dos modelos. Nesta aplicação, optou-se por mostrar modelos simples, com menos informação, sem pormenores como sombras, pois isso implicaria ficheiros maiores guardados na memória do telemóvel e também mais poder de processamento necessário para os representar. Os modelos usados não têm, portanto, tanto realismo, mas considerou-se serem suficientes para cumprir o objetivo.

O objetivo chave da visualização é que o médico consiga perceber onde está a área a estimular. Para isso, e para que se distingam também as diferentes áreas umas das outras, a aplicação desenha cada área com uma cor, sendo que a cor da área a estimular se destaca das restantes. Na versão atual, as restantes áreas estão coloridas numa gama de cinzentos, para permitir que as áreas se distingam entre si mas que o foco esteja na área a estimular (ver Figura 5.15).

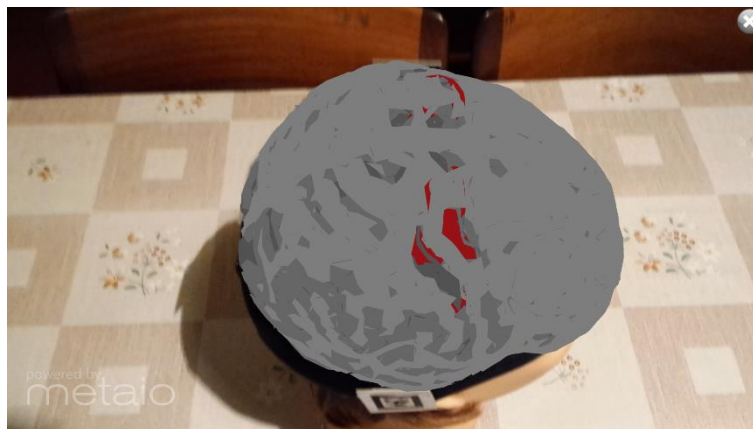


Figura 5.15 - Modelo virtual do cérebro com várias áreas distinguidas pela cor, com uma cor mais distintiva para a área a estimular

Nas secções seguintes descrevem-se dois problemas estudados ao longo do desenvolvimento do projeto: a perceção de profundidade e o uso de estereoscopia.

5.4.1 Perceção de Profundidade

Um dos principais problemas com as aplicações de RA é a perceção de profundidade. Os modelos virtuais são desenhados sobre a imagem da câmara, pelo que aparentam estar por cima de todos os elementos reais visualizados. Em particular nesta

aplicação, o modelo virtual irá aparecer sempre sobre a cabeça do paciente, como se pretende, para que o médico possa integrar as duas imagens e que o modelo virtual indique com sucesso a posição e orientação do cérebro dentro da cabeça do paciente. Contudo, o problema surge quando elementos reais se sobrepõem à cabeça do paciente, pois vão ser mostrados por trás do modelo virtual do cérebro, desfazendo a percepção de que o cérebro virtual está na cabeça do paciente. Mas durante a aplicação da técnica de EMT o médico usa um aparelho de estimulação sobre a cabeça do paciente, que muito provavelmente ao longo do uso vai estar colocado entre a cabeça do paciente e o modelo virtual, impedindo a correta percepção da profundidade do modelo virtual.

Uma forma de abordar este problema, é retirar foco à imagem da câmara. É possível fazer isso com o Metaio SDK, pois este permite adicionar um fundo sobre a imagem da câmara, com cor e transparência personalizável. Assim, pode-se simplesmente tapar a imagem da câmara com um fundo opaco sobre o qual aparecem os modelos virtuais ou então retirar mais ou menos foco, dando transparência ao fundo para que se consiga ver a imagem da câmara para ter noção do espaço envolvente mas diminuindo a interferência que alguns elementos da imagem podem provocar.

Porém, esta abordagem só por si não é ideal, pois evita problemas de percepção de profundidade mas não os resolve. Além disso, no contexto da EMT é essencial que o médico consiga ver não só a área a estimular mas também onde se encontra a sua mão e o aparelho de estimulação relativamente a essa área, para que possa direcionar o aparelho até à área pretendida. Assim, se a grande visibilidade da mão do utilizador na imagem da câmara estava a dificultar a percepção de profundidade, a sua redução de visibilidade, por outro lado, faz com que o médico não saiba onde se encontra a sua mão e que área do cérebro está no momento a estimular.

Para resolver ambos os problemas, decidiu-se criar um modelo virtual do aparelho de estimulação, para acompanhar os movimentos do aparelho real, na mão do utilizador, o que inclusive melhora a percepção da distância entre o cérebro do paciente e a mão do médico. Para tal, é preciso fazer o *tracking* do aparelho de estimulação, colocando uma marca no mesmo. Este modelo virtual do aparelho de estimulação, que designamos por glifo, não necessita representar com fidelidade a forma do aparelho usado mas serve como um guia visual para a posição e orientação do aparelho.

Foram produzidos dois modelos Blender para este glifo (ver Figura 5.16). O primeiro, designado por Coil, é baseada no formato real de um aparelho usado na EMT, numa versão simplificada. O segundo, designado por Arrow, consiste num tetraedro com base isósceles e altura relativamente grande, para que o vértice que não pertence à base se destaque e o polígono se assemelhe a uma seta, indicando a orientação do aparelho de estimulação. O utilizador pode escolher entre os dois glifos na interface da aplicação (ver secção 5.6).

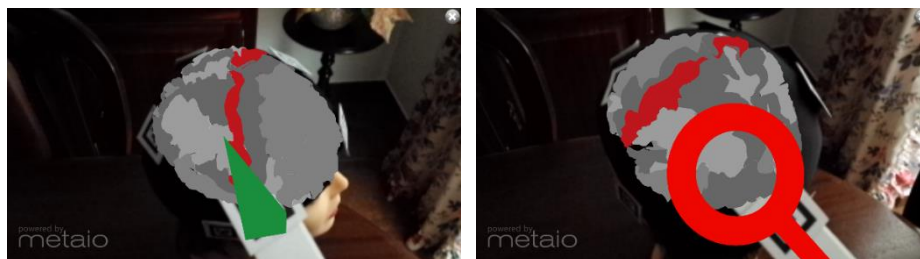


Figura 5.16 - Glifo do aparelho de estimulação nas duas representações: Arrow (esquerda) e Coil (direita)

Como referimos, o glifo melhora a perceção de profundidade, fazendo com que a visualização se centre nos elementos virtuais, que contêm toda a informação necessária para a aplicação da técnica de EMT. Uma das informações mais importantes que a visualização tem de dar ao utilizador é a relação espacial em cada instante entre o cérebro do paciente e o aparelho de estimulação. Com modelos virtuais para o cérebro e para o aparelho representados no ecrã, essa relação fica clara, mas informação adicional sobre a distância a que os dois elementos se encontram um do outro ajudaria ainda mais o utilizador. Essa distância obtém-se facilmente, graças às marcas que existem tanto na cabeça do paciente como no aparelho de estimulação. Para não sobrecarregar a interface de utilizador com um número a indicar a distância, optou-se por uma representação mais visual e associou-se a distância à cor do glifo. Assim, o glifo é vermelho quando o aparelho está distante do cérebro, amarelo quando a uma distância intermédia e muda para verde quando se encontra próximo do cérebro do paciente, indicando que está perto o suficiente para que a estimulação consiga produzir efeito (Figura 5.17).

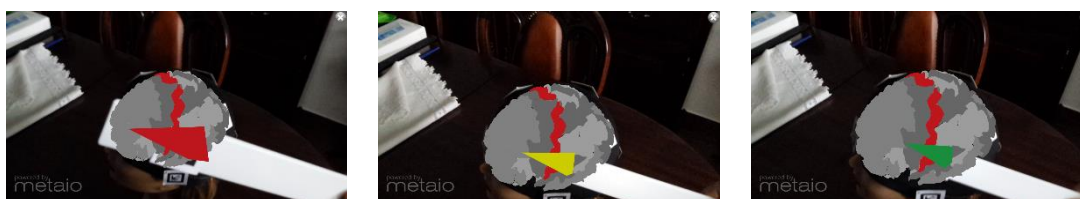


Figura 5.17 – Ecrã do modo de RA com o glifo do aparelho de estimulação das várias cores, consoante a distância deste ao modelo do cérebro: distantes (esquerda), a distância intermédia (centro) e próximos (direita)

Outra questão relevante para a percepção de profundidade e um alinhamento correto do modelo do cérebro com a cabeça do paciente é o tamanho do modelo. O Metaio SDK ajusta automaticamente o tamanho do modelo virtual em função da distância a que se encontram as marcas. Para que o modelo apareça com o tamanho correto é portanto necessário que a aplicação tenha informação correta sobre o tamanho real das marcas, informação essa que é dada no ficheiro XML de configuração do *tracking*. Para além disso, é apenas preciso que o modelo guardado na memória do telemóvel contenha as dimensões reais do cérebro do paciente, o que acontecerá à partida, pois o modelo é obtido a partir de ressonância magnética realizada ao cérebro do próprio paciente.

A percepção da profundidade pode ainda ser melhorada com o uso de estereoscopia, que foi uma hipótese em estudo durante este projeto.

5.4.2 Estereoscopia

Estereoscopia é uma técnica para criar a ilusão de profundidade numa imagem através da associação a cada olho de uma versão da mesma imagem, vista de uma perspetiva ligeiramente diferente. Esta técnica é cada vez mais comum nos dias de hoje e a tecnologia tem avançado de forma a facilitar a sua aplicação, mas ainda assim esta técnica continua a precisar do uso de algum equipamento especializado para produzir o seu efeito. Um exemplo comum são ecrãs que representam a imagem vista das duas perspetivas lado a lado e graças a óculos próprios cada olho recebe apenas uma das imagens, fazendo com que as duas se combinem, no cérebro, dando a percepção de profundidade tridimensional.

O uso desta técnica no contexto do projeto foi estudado e fazia parte do plano inicial, pois trazia algumas vantagens. Em primeiro lugar, permitia melhorar a percepção de profundidade da visualização. Em segundo lugar, o *smartphone* teria de ser colocado num suporte próprio para o efeito, que o médico teria de usar como se fossem óculos de RA, pelo que ficava com ambas as mãos livres para executar a técnica de EMT.

No que diz respeito a este tipo de suporte, trata-se de equipamento próprio para experiências de Realidade Aumentada ou Virtual usando *smartphones* como ecrã e unidade de processamento. O *smartphone* coloca-se na parte da frente dos óculos, ocupando o campo de visão do utilizador. Este passa a ver a realidade através da câmara

do telemóvel, imagem à qual são sobrepostos os elementos de RA. O suporte escolhido para testar durante o desenvolvimento do projeto foi o Durovis Dive, como referido na secção 4.2.2. Este possui uma divisória a meio, entre os dois olhos, para que cada olho veja apenas a sua metade do ecrã. Para se obter o efeito estereoscópico, as aplicações devem, portanto, mostrar em cada lado do ecrã duas perspetivas diferentes da mesma imagem.

Conseguiu-se perceber como obter uma visão estereoscópica dos modelos virtuais usando o Metaio SDK. Existem métodos já implementados para esse efeito, dividindo o ecrã ao meio verticalmente e mostrando os modelos vistos de perspetivas ligeiramente diferentes em cada lado, simulando a visão humana (Figura 5.18). É possível ajustar as perspetivas para melhor corresponderem às posições dos olhos do utilizador relativamente à câmara.



Figura 5.18 - Visualização de um modelo 3D com o ecrã dividido ao meio

Esta visualização estereoscópica com perspetivas diferentes transmitidas a cada olho, porém, é aplicada apenas aos modelos virtuais, enquanto a imagem da câmara se mantém, duplicada, igual para os dois olhos, uma vez que o *smartphone* possui apenas uma câmara, o que prejudica a visualização. Por esta razão, usualmente quando se utiliza o modo estereoscópico do Metaio SDK a imagem da câmara não é representada, sendo substituída por um fundo de uma cor personalizável. No caso desta aplicação, parece não fazer sentido ocultar completamente a imagem da câmara, pois isso retira ao médico quase toda a informação visual de referência para ele se guiar no espaço. Felizmente, é possível chegar a um compromisso, alterando a transparência do fundo que é colocado sobre a imagem da câmara, criando uma visualização que continua a fornecer ao médico referências espaciais, através da imagem da câmara, mas que retira foco dessa imagem, para que ele se concentre nas representações virtuais estereoscópicas criadas (Figura 5.19).

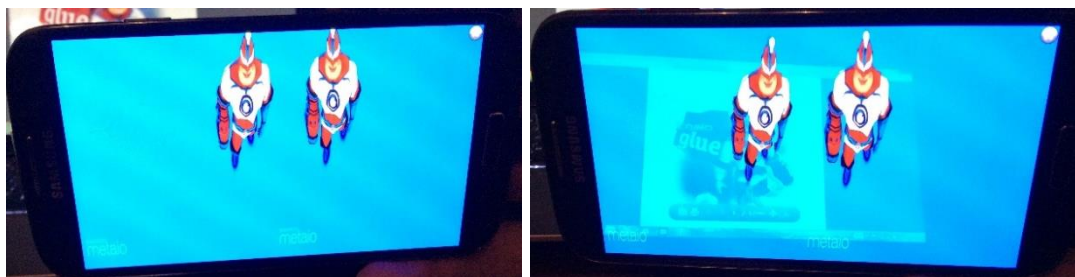


Figura 5.19 - Visualização de um modelo 3D com fundo colorido: opaco (esquerda); com transparência (direita)

Um modo de RA em visualização estereoscópica foi adicionado à aplicação, funcionando da mesma forma que o modo de RA normal mas com o ecrã dividido, com as características descritas acima, para ser usado com os óculos especiais. No entanto, após se terem feito algumas experiências com o uso de estereoscopia, não foi possível chegar a uma visualização agradável, pelo que se optou por não incluir esse modo na aplicação.

5.5 Modo de Realidade Virtual

O modo de RV incluído na aplicação (e que não estava previsto no início do projeto) permite interagir de forma táctil com o modelo virtual do cérebro, rodando-o e aumentando ou diminuindo a sua escala, sem que este esteja associado à imagem captada pela câmara da cabeça do paciente. Neste modo, o mesmo modelo do cérebro do paciente é usado, com as mesmas cores associadas às várias áreas, mas representado num referencial no centro do ecrã.

Como explicado antes, existem duas opções de interação com o modelo: *zoom-in/out* e rodar. Para aumentar ou diminuir o modelo foi implementado o gesto de *pinch zoom*, que consiste em colocar dois dedos no ecrã do telemóvel e afastá-los, para aumentar a escala do modelo, ou aproximá-los, para diminuir essa escala. Para rodar o modelo o utilizador deve colocar também dois dedos no ecrã e movê-los circularmente, num sentido ou noutro, para que o modelo rode num ou noutro sentido em torno de um eixo de rotação determinado.

Esse eixo de rotação em torno do qual o modelo roda pode ser alterado pelo utilizador através de dois botões na interface da aplicação, correspondentes ao eixo horizontal do ecrã e ao eixo vertical do ecrã. Não se considerou dar a opção de rotação

em torno do eixo perpendicular ao ecrã, porque com este tipo de rotações ficaria sempre visível a mesma projeção do modelo.

Também se pode rodar automaticamente o modelo para o conseguir visualizar segundo seis vistas predeterminadas: frente, trás, esquerda, direita, cima e baixo. A Figura 5.20 mostra a interface de utilizador da aplicação no modo de RV.

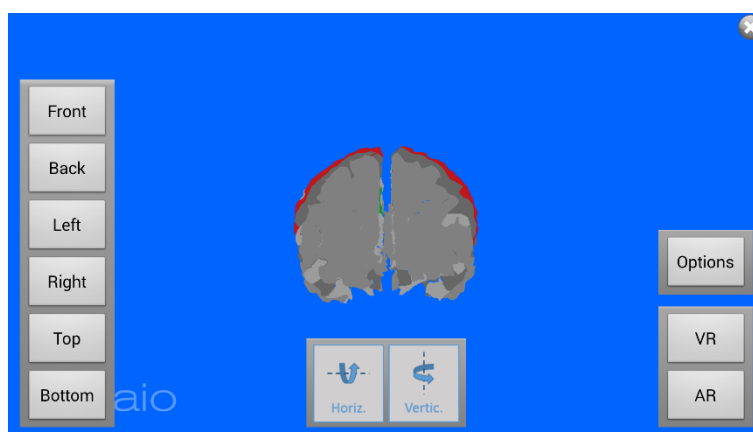


Figura 5.20 - Ecrã do modo de RV, com todos os botões da interface visíveis

5.6 Visualização Seletiva

Na sequência da implementação do modo de Realidade Virtual, em que o utilizador pode interagir por toque com o modelo virtual do cérebro, chegou-se à conclusão que seria também bastante útil permitir ao utilizador esconder algumas das áreas do cérebro e visualizar apenas as que quisesse, até porque algumas áreas não superficiais apenas conseguiriam ser visualizadas dessa maneira. Então, uma funcionalidade a que se chamou “Visualização Seletiva” foi implementada na aplicação, funcionando tanto no modo de RA como no modo de RV e permitindo ao utilizador escolher exatamente que áreas do cérebro a aplicação deve desenhar.

Para que essa escolha fosse feita, um menu de opções foi criado para o efeito (menu ao qual mais tarde se adicionou também a opção de escolha do glifo utilizado para representar o aparelho de estimulação no modo de RA). Neste menu, existe uma caixa de verificação (*checkbox*) para cada área do cérebro, que o utilizador pode marcar ou desmarcar para tornar essa área visível ou invisível, respetivamente. Acima das caixas de verificação para cada área, porém, existem mais algumas caixas relativas a conjuntos específicos de áreas do cérebro, em particular os quatro lobos – frontal, parietal, temporal

e occipital – e os dois hemisférios – esquerdo e direito. Ao marcar/desmarcar a caixa de verificação de alguma dessas regiões, as caixas de verificação de todas as áreas que pertencem a essa região ficam automaticamente marcadas/desmarcadas, respetivamente, permitindo ao utilizador controlar mais facilmente a visibilidade geral por lobos ou hemisférios do cérebro. O utilizador pode confirmar a sua escolha premindo o botão “OK”, que o leva de volta ao ecrã do modo da aplicação em que se encontrava. Os ecrãs do menu de opções e do modo de RV com visualização seletiva podem ser observados na Figura 5.21.

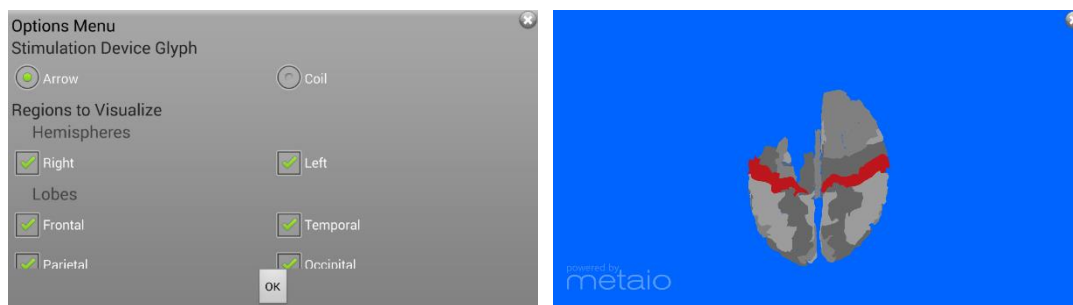


Figura 5.21 - Ecrã do menu de opções (esquerda) e modo de RV com visualização seletiva (direita)

Para que esta funcionalidade fosse possível, a aplicação precisava de reconhecer individualmente cada área do cérebro – o que já acontecia, pois cada área correspondia a um sub-modelo diferente do modelo virtual do cérebro – mas mais que isso precisava de saber a que lobo e hemisfério cada área pertencia. Essa informação tinha portanto de ser incluída de alguma forma na aplicação, o que se fez através de um ficheiro de texto que já existia onde se indicavam os caminhos para os ficheiros que guardavam os modelos tridimensionais das várias áreas do cérebro. Mais informação sobre este ficheiro de texto é dada na secção 5.7.3. Mais informação sobre a forma como se dividiram pelos lobos as áreas consideradas ao longo do desenvolvimento do projeto é dada na secção 5.7.2.

5.7 Aquisição e Processamento dos Modelos do Cérebro

5.7.1 Obtenção dos modelos

Uma das questões essenciais para o sucesso do projeto prendia-se com a obtenção e preparação do modelo virtual 3D do cérebro. Graças ao Professor Hugo Ferreira foi possível perceber como funciona o processo de criação dos modelos virtuais a partir de ressonâncias magnéticas, pela experiência que ele já tem nesses procedimentos. Mais

tarde, teve-se acesso à tese de um aluno orientado por este professor, o Ricardo Ribeiro, que também trabalhou com modelos do cérebro, apesar de não os ter usado em aplicações de Realidade Aumentada. Mais concretamente, a sua tese tratava do “Desenvolvimento de uma interface de visualização de conectividade cerebral usando realidade virtual e controlo por gestos” [41]. Em seguida descreve-se o processo pelo qual as imagens obtidas por ressonância magnética foram, naquele caso, convertidas num modelo 3D do cérebro.

1. As imagens chegaram do serviço de imagiologia em formato DICOM, ponderadas em T1;
2. Foi feita uma conversão das imagens para o formato NIFTI, numa matriz de intensidade de sinal;
3. Com o *software* Freesurfer [42] fez-se uma segmentação do cérebro em substância cinzenta, branca e líquido cefalorraquidiano, tendo por base dois mapas anatómicos, com uma divisão entre as regiões corticais e as subcorticais;
4. Realizou-se a parcelização do cérebro, com base em atlas anatómicos (ver secção 5.7.2), obtendo-se um ficheiro para cada hemisfério e atlas usado, onde estava contida a parcelização cerebral, e os respetivos ficheiros com as informações anatómicas relevantes;
5. Estes ficheiros apenas podiam ser lidos por *software* de imagem médica, pelo que era necessário fazer uma conversão para um formato mais comum, que pudesse ser lido por *software* de modelação, mais acessível, e pela aplicação no telemóvel. Através de um projeto chamado Brainder [38], criado por A. M. Winkler, e usando ferramentas desenvolvidas pela Universidade de Oxford (referidas em [41]), conseguiu-se fazer essa conversão, tendo sido criado um *script* em MATLAB [43] para autonomizar o processo. Para além da conversão de formato, separaram-se as áreas da parcelização da superfície cortical em ficheiros individuais, para poderem ser visualizados separadamente. O formato do ficheiro passou a ser o OBJ (extensão “.obj”).

A parcelização referida acima consiste em individualizar cada área do cérebro e permite que cada uma seja tornada, como vimos no processo descrito acima, num modelo 3D individual. Essa individualização é fulcral para o funcionamento correto da aplicação, pois é o que permite atribuir cores diferentes a cada área e, em particular, destacar a área

que se pretende estimular, para além de possibilitar a funcionalidade de Visualização Seletiva. É importante ressaltar que no contexto desta aplicação apenas se pretendem visualizar as áreas do córtex cerebral, pelo que parcelizações de regiões subcorticais não são necessárias.

A parcelização do córtex cerebral é efetuada, como referido, com base em atlas anatómicos, que definem por que áreas unitárias é organizado o cérebro. Assim, a divisão em várias áreas visíveis e reconhecidas individualmente na aplicação varia em função do atlas escolhido pelo médico no *software* Freesurfer aquando da criação dos modelos 3D. Existem vários atlas usados pelo Freesurfer, em particular:

- o atlas cortical de Destrieux, desenvolvido por Destrieux et al. [44], que faz uma divisão de cada hemisfério em 74 áreas;
- o atlas cortical de Desikan-Killiany (DK), desenvolvido por Desikan et al. [45], que faz uma divisão de cada hemisfério em 34 áreas;
- o atlas cortical de Desikan-Killiany-Tourville (DKT), adaptação do atlas DK desenvolvida por Klein e Tourville [46], eliminando 3 áreas do atlas original, para uma divisão de cada hemisfério em 31 áreas.

No Anexo D podem ser consultadas as listas das áreas de cada um destes atlas, bem como esquemas da organização dessas áreas.

No final deste processo, obtêm-se ficheiros de formato OBJ (com extensão “.obj”), cada um deles correspondente ao modelo 3D de uma das áreas do cérebro pretendidas. O formato OBJ é um formato de ficheiro de definição de geometrias, criado inicialmente pela empresa Wavefront Technologies e adotado por muitos fornecedores de aplicações de gráficos 3D, sendo hoje universalmente aceite e utilizado. Os ficheiros OBJ conseguem ser importados para o Blender [6], para mais fácil visualização e edição, e também carregados pelo Metaio SDK como geometrias para visualização dentro da aplicação, como pretendido. O Blender permite exportar objetos em formato OBJ, pelo que é possível importar e editar os modelos virtuais no Blender antes de os exportar e colocar na memória do telemóvel para serem carregados pela aplicação.

Todos os modelos referidos e usados neste projeto, bem como os que podem ser obtidos pelo processo acima, são superfícies definidas por malhas poligonais.

5.7.2 Modelos usados e sua organização

Ao longo do desenvolvimento do projeto foram usados modelos virtuais 3D obtidos da internet, fornecidos pelo projeto Brainder [38]. Esses modelos foram obtidos seguindo o processo descrito na secção anterior, a partir de imagens de ressonância magnética adquiridas no Research Imaging Institute da Universidade do Texas.

O projeto Brainder oferece vários conjuntos de modelos, correspondentes a parcelizações da região cortical com os três atlas referidos na secção anterior: Destrieux, DK e DKT. Contudo, neste projeto, todas as parcelizações criam um modelo correspondente a uma área adicional, chamada “Unknown”, constituída pela parte do córtex cerebral que não foi possível associar a nenhuma das áreas definidas por cada atlas. O projeto Brainder oferece também um conjunto não parcelizado, com um modelo para cada hemisfério, apenas.

Além disso, existem três variantes dos modelos: uma correspondente à camada exterior do córtex de cada área (*pial*), outra correspondente à camada interior do córtex de cada área (*white*) e outra correspondente a versões infladas de cada área, permitindo ver de uma forma mais clara e simples as ligações entre as regiões, apesar de cada área ficar ligeiramente distorcida (*inflated*).

O projeto Brainder disponibiliza ainda modelos virtuais para as várias estruturas subcorticais do mesmo cérebro, caso desejado. Na Figura 5.22 podem observar-se imagens ilustrativas do modelo do cérebro com as parcelizações segundo os três atlas e nas três variantes do modelo.

Todos os conjuntos de modelos foram transferidos e observados, tendo um deles sido usado na aplicação durante o resto do desenvolvimento do projeto. O conjunto escolhido foi o parcelizado pelo atlas DK na variante *white*, correspondente à camada interior do córtex cerebral. No entanto, qualquer conjunto de modelos pode ser usado mais tarde, consoante a vontade do médico que introduza os dados na memória do *smartphone*.

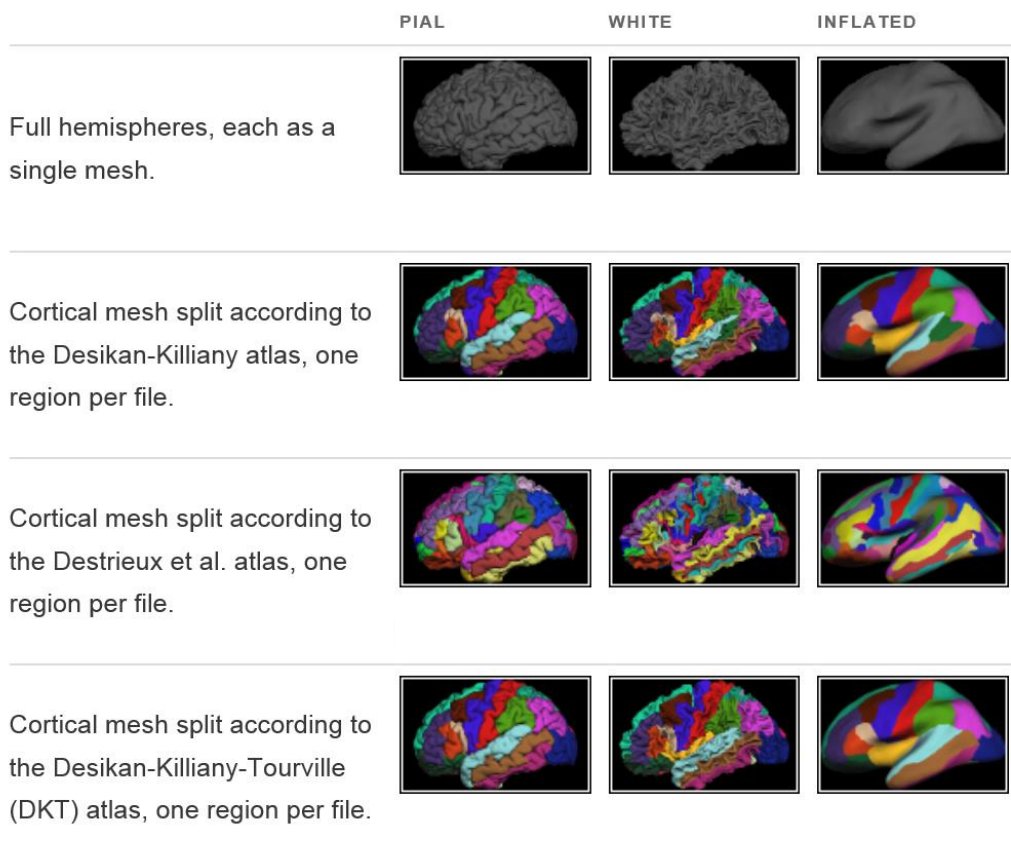


Figura 5.22 - Imagens ilustrativas das parcelizações de acordo com os três atlas e nas três variantes (versão modificada da imagem apresentada na página do projeto Brainder [38])

A informação sobre que modelos deve a aplicação carregar é dada num ficheiro de texto (ficheiro de definição das geometrias) que deve ser criado e guardado no telemóvel, como explicado na secção 5.7.3. Nesse ficheiro deve ser também associado a cada uma das áreas o lobo a que pertence, para permitir à funcionalidade de Visualização Seletiva a seleção e visualização de áreas agrupadas por lobos. No caso dos modelos do projeto Brainder, em particular do conjunto de modelos parcelizado pelo atlas DK na variante *white*, que foi o conjunto usado com a aplicação durante a maior parte do período de desenvolvimento do projeto, existiram algumas dificuldades na divisão de todas as áreas pelos 4 lobos.

A informação na página [47] permitiu associar a maior parte das áreas a um dos lobos, mas duas áreas restaram: a “Insula” e a área adicional “Unknown”. De acordo com [48], a insula é um quinto lobo do cérebro humano. Já Klein e Tourville [46], no estudo que fazem do atlas DKT, juntam a insula ao córtex cingulado (que é composto por 4 áreas) num conjunto de áreas fora dos 4 lobos. As 4 áreas do córtex cingulado já haviam sido divididas pelos lobos de acordo com [47], duas para o lobo frontal e outras duas para o

parietal. Assim, e para manter a simplicidade da divisão do córtex cerebral nos 4 lobos, decidiu-se incluir a insula como área do lobo temporal, por serem adjacentes um ao outro. Em relação à área “Unknown”, optou-se por se adicionar temporariamente ao lobo occipital, por ser o mais pequeno e com menos áreas. No Anexo E pode ver-se a divisão que foi efetuada das áreas do atlas DK pelos 4 lobos do córtex cerebral.

Estas divisões não são definitivas, pois cabe ao médico que criar o ficheiro de definição das geometrias com os dados do seu paciente decidir a que lobo associar cada um dos modelos que obtiver com o processo de parcelização. Como trabalho futuro na aplicação, será possível, se médicos julgarem necessário, permitir uma visualização seletiva em lobos considerando a insula como um quinto lobo, isolando também o córtex cingulado dos lobos ou mesmo adicionando outros conjuntos de áreas, como as áreas subcorticais.

5.7.3 Ficheiro de texto de definição das geometrias

Um dos objetivos fundamentais da aplicação é que esta consiga realçar, na representação do cérebro, uma ou mais áreas de interesse, por exemplo para estimular durante o procedimento de EMT. Para isso, a aplicação precisa de saber distinguir as várias áreas do cérebro, de acordo com uma divisão desejada pelo médico. Com a introdução da funcionalidade de Visualização Seletiva esta necessidade foi reforçada, uma vez que era preciso distinguir todas as áreas e poder alterar as propriedades visuais da representação de cada uma em tempo de execução. Para isto ser possível, era necessário que cada área do cérebro correspondesse a um modelo virtual diferente (ou geometria, como a documentação do Metaio SDK costuma designar), ou seja, que cada área fosse representada por um modelo guardado num ficheiro diferente. Como vimos atrás, o processo de obtenção dos modelos 3D permite esta estruturação, sendo que é durante esse processo que o médico tem de definir qual a divisão em áreas que quer aplicar ao córtex cerebral, podendo usar um de vários atlas de parcelização definidos.

É possível então obter um conjunto de ficheiros OBJ, cada um guardando o modelo virtual de uma das áreas do córtex cerebral a representar. Esses ficheiros devem ser guardados na memória do *smartphone* (memória interna ou externa, num cartão SD, por exemplo) para que a aplicação os consiga ler. Contudo, é necessário indicar à aplicação exatamente quais os ficheiros que deve carregar e onde estão guardados.

Além disso, tem de ser associada alguma informação a cada área, como a cor com que esta deve ser representada. No que diz respeito a este processo de colorir as áreas, baseia-se em aplicar a cada área uma textura correspondente a uma imagem de uma cor lisa. Esta aplicação de texturas está programada no código da aplicação e texturas de várias cores já foram criadas e estão guardadas em ficheiros de imagem PNG que estão incluídos na aplicação. O utilizador tem de definir, porém, que cor associar a cada área.

Para além da cor, é preciso associar a cada área também o hemisfério e lobo a que esta pertence, para permitir a visualização seletiva por estes conjuntos de áreas (hemisférios e lobos) e não só por áreas individuais.

Para resolver todos estes problemas, foi implementada a leitura de um ficheiro de texto (extensão “.txt”) que o médico deve criar e guardar no telemóvel, ficheiro esse a que se deu o nome de “ficheiro de definição das geometrias”. Na primeira linha do ficheiro de texto devem ser colocados alguns números, separados por espaços, para auxiliar a aplicação na leitura do ficheiro e a guardar os dados: número total de áreas, número de áreas no hemisfério esquerdo e no direito e número de áreas em cada lobo: frontal, temporal, parietal e occipital. A partir daí, cada uma das restantes linhas do ficheiro corresponde a uma das áreas e contém a seguinte informação, separada por espaços:

- Caminho relativo na memória do telemóvel do ficheiro que contém o modelo da área;
- “L”, se a área pertencer ao hemisfério esquerdo, ou “R”, se a área pertencer ao hemisfério direito;
- “F”, se a área pertencer ao lobo frontal, “T”, se a área pertencer ao lobo temporal, “P”, se a área pertencer ao lobo parietal, ou “O”, se a área pertencer ao lobo occipital;
- Cor pretendida para aplicar à área.

Mais informação sobre os caminhos de ficheiro que devem ser usados pode ser encontrada na secção 5.8. Em relação às cores, várias estão disponíveis para serem usadas já pela aplicação:

- Grey25, Grey37,5 e Grey50, três tons diferentes de cinzento para, por exemplo, colorir a generalidade das áreas numa escala de cinzentos;
- Red, Red1 e Red2, três tons diferentes de vermelho para, por exemplo, colorir a generalidade das áreas numa escala de vermelhos ou então usar um dos vermelhos para colorir as áreas a estimular;
- Green, Blue, Cyan, Magenta e Yellow para colorir áreas a verde, azul, ciano, magenta e amarelo, respetivamente.

Na Figura 5.23 vê-se um excerto de um ficheiro de definição das geometrias usado durante o desenvolvimento do projeto.

```

70 35 35 26 20 14 10
white_DK_obj_dec/lh.white.DK.bankssts.obj L T Grey25
white_DK_obj_dec/lh.white.DK.caudalanteriorcingulate.obj L F Grey25
white_DK_obj_dec/lh.white.DK.caudalmiddlefrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.cuneus.obj L O Grey37,5
white_DK_obj_dec/lh.white.DK.entorhinal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.frontalpole.obj L F Grey50
white_DK_obj_dec/lh.white.DK.fusiform.obj L T Grey25
white_DK_obj_dec/lh.white.DK.inferiorparietal.obj L P Grey25
white_DK_obj_dec/lh.white.DK.inferiortemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.insula.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.isthmuscingulate.obj L P Grey50
white_DK_obj_dec/lh.white.DK.lateraloccipital.obj L O Grey50
white_DK_obj_dec/lh.white.DK.lateralorbitofrontal.obj L F Grey25
white_DK_obj_dec/lh.white.DK.lingual.obj L O Grey25
white_DK_obj_dec/lh.white.DK.medialorbitofrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.middletemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.paracentral.obj L F Grey50
white_DK_obj_dec/lh.white.DK.parahippocampal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.parsopercularis.obj L F Grey25
white_DK_obj_dec/lh.white.DK.parsorbitalis.obj L F Grey25
white_DK_obj_dec/lh.white.DK.parsstriangularis.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.pericalcarine.obj L O Grey37,5
white_DK_obj_dec/lh.white.DK.postcentral.obj L P Red
white_DK_obj_dec/lh.white.DK.posteriorcingulate.obj L P Grey50
white_DK_obj_dec/lh.white.DK.precentral.obj L F Grey50
white_DK_obj_dec/lh.white.DK.precuneus.obj L P Grey25

```

Figura 5.23 - Excerto de um ficheiro de definição das geometrias usado durante o desenvolvimento do projeto

As geometrias definidas com o ficheiro mostrado na Figura 5.23 correspondem às áreas definidas pelo atlas DK na variante *white*. Como explicado na secção 5.7.1, o atlas DK considera 34 áreas diferentes para cada hemisfério. Contudo, os modelos em questão, retirados do projeto Brainder [38], incluem um modelo adicional chamado “Unknown”, correspondente à parte do córtex cerebral que não foi associada a nenhuma das outras áreas, subindo o número de áreas para 35 por hemisfério.

No exemplo deste ficheiro, foi escolhida a área pós-central do cérebro (“Postcentral”) como a área a destacar, tendo-lhe sido associada uma textura vermelha.

As texturas associadas às restantes áreas variam entre três tons de cinzento. O resultado de tal atribuição de cores pode ser visualizado na Figura 5.24.

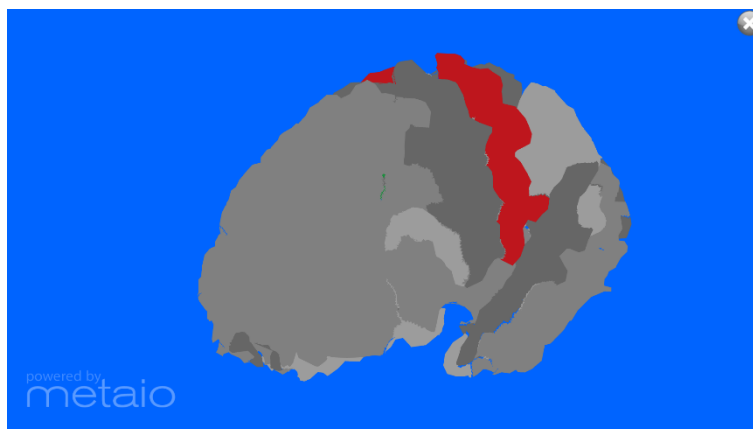


Figura 5.24 - Ecrã do modo de RV apresentando o cérebro colorido de acordo com o ficheiro de definição de geometrias exemplificado

No que toca à divisão por lobos, algumas complicações existiram, como se encontra explicado na secção 5.7.2. A divisão a que se chegou encontra-se explicitada no Anexo E e a versão completa do ficheiro de texto encontra-se no Anexo F.

5.7.4 Teste de Carga

Após ter o protótipo da aplicação concretizado, revelou-se pertinente estudar algumas características da visualização, em particular o desempenho da aplicação em termos de *rendering* dos modelos tridimensionais.

Os modelos virtuais são formados por polígonos, geralmente triângulos, dispostos numa malha poligonal, sendo que um maior número de polígonos num modelo confere-lhe uma maior resolução, ou seja, mais informação no mesmo objeto. Em imagiologia médica, é bastante útil que estes modelos tenham muita definição, para que a representação do cérebro do paciente seja bastante fidedigna, uma vez que estas imagens são usadas para fins muito importantes para a saúde do paciente. Assim, os modelos tridimensionais do cérebro obtidos por ressonância magnética têm uma resolução muito elevada, ou seja, um elevado número de polígonos.

Infelizmente, o poder de computação presente na maioria dos *smartphones* atuais ainda não permite um *rendering* de modelos com um tão grande número de polígonos num intervalo de tempo suficientemente pequeno para que a aplicação que pretendemos

funcione em tempo real, isto é, gerando os modelos virtuais as vezes necessárias para acompanhar os movimentos da câmara e as mudanças captadas por esta, mantendo-os sempre alinhados com a cabeça dos pacientes.

Para conseguir bons desempenhos na aplicação, foi então necessário reduzir o número de informação que os modelos continham, diminuindo o número de polígonos. Para isto, usou-se um modificador “*Decimate*” no Blender. Este modificador aplica alguns algoritmos que permitem ir agrupando polígonos adjacentes num só, reduzindo o número de polígonos total do modelo, sendo possível definir a razão que se quer usar no processo, isto é, quanto se quer reduzir no número de polígonos. Este processo foi efetuado em alguns dos modelos com que se trabalhou, conforme se achou necessário. A razão aplicada no modificador foi 0.1, ou seja, o número de polígonos destes modelos diminuiu para aproximadamente um décimo do número original. Com este processo, os modelos inicialmente mais ricos perderam algum do detalhe que tinham, mas constatou-se que a resolução final ainda é suficiente para se conseguir uma visualização detalhada do cérebro dos pacientes.

Por todas estas considerações, revelou-se bastante útil fazer um estudo do desempenho da aplicação em função dos modelos carregados, do número de polígonos que continham, e também de outras características da visualização, como por exemplo a aplicação de diversas texturas e também de transparências. Para tal, mediu-se o número de vezes que o *smartphone* conseguia fazer o *rendering* dos modelos num segundo, ou seja, o número de *frames-per-second* (FPS). Estes dados foram retirados a apontar para o modelo de cabeça feminina com cinco marcas, pois era esta a configuração de marcas que se usava na altura. Nessas experiências, cada cenário de teste foi testado 20 vezes, tendo sido feita a média dos resultados obtidos, que é apresentada na tabela. Para além das FPS de *rendering*, apresentam-se também as FPS do *tracking*, isto é, quantas vezes por segundo a aplicação deteta as marcas e o centro do referencial, e as FPS da câmara, ou seja, quantas vezes por segundo a imagem da câmara é capturada para ser analisada.

Relativamente aos modelos usados neste teste, usámos modelos provenientes de duas fontes diferentes. Um conjunto de modelos foi-nos providenciado pelo Professor Hugo Ferreira, sendo composto por três modelos relativos a três camadas diferentes do cérebro mais um modelo relativo a uma região específica do cérebro, para representar a

área a estimular. Outro conjunto de modelos foi retirado da web, a partir do projeto Brainder [38], que fornece gratuitamente modelos de alta qualidade do cérebro, segundo várias parcelizações, como explicado na secção 5.7.2.

Relativamente aos modelos do Professor Hugo, coloriu-se cada uma das três camadas do cérebro com uma textura diferente, tendo-se testado três combinações diferentes de texturas: uma RGB (uma camada vermelha – “Red”, uma camada verde – “Green” e uma camada azul – “Blue”), uma escala de cinzentos e uma escala de vermelhos. Com a escala de vermelhos, testou-se ainda uma visualização com transparência nas camadas. Em todos estes quatro casos, a área a estimular tinha uma textura verde, para dar destaque. Fez-se ainda um caso de teste em que apenas se visualizou a área a estimular, com a textura verde, o que constituiu o caso com menor número de polígonos para desenhar.

No caso dos modelos do projeto Brainder, foram usados os modelos correspondentes a uma divisão simples do cérebro nos dois hemisférios, na versão original não decimada e também na versão decimada, com aproximadamente um décimo do número de polígonos original. A cada hemisfério foi aplicada uma textura de uma cor diferente (verde e azul). Para um último caso de teste foi usado um conjunto de modelos relativos a uma parcelização do cérebro em várias áreas, de acordo com o atlas Desikan-Killiany (DK). Neste último cenário não se usaram todos os modelos, apenas 28 dos 70 (14 de 35 para cada hemisfério), para não exceder o número de polígonos suportável pelo *smartphone*. Para se representarem todas as áreas, teria de ser aplicado o modificador “Decimate” a cada um dos modelos, o que só foi efetuado depois da realização deste teste, graças a um script Python que se criou, como explicado na secção 5.7.5. Neste último caso, as áreas foram representadas com texturas vermelhas, verdes ou azuis.

Na Tabela 4 podem observar-se os resultados, ordenados pelo número de polígonos desenhados em cada caso de teste. Observando estes resultados, conclui-se que as FPS do *rendering* variam, de facto, de acordo com o número total de polígonos que se está a tentar representar. Com cerca de 10.000 polígonos o desempenho ronda as 60 fps, com 60.000 polígonos o desempenho varia entre as 40 e as 55 fps, com 220.000 polígonos o número altera-se pouco, para aproximadamente 40 fps, mas quando o número sobe para 580.000 polígonos então o desempenho baixa drasticamente para cerca de 5 fps. Neste

caso, a aplicação inclusive envia uma mensagem para a consola do Android, informando da má execução da aplicação: “*Skipped 32 frames! The application may be doing too much work on its main thread.*” De facto verifica-se que a aplicação fica bastante lenta, inclusive a imagem da câmara fica parada ou demora muito tempo a atualizar. Isto ocorre porque o processador está demasiado ocupado a fazer o *rendering* dos modelos e não consegue executar o resto do trabalho que devia. Este é também o único caso em que o número de fps do *tracking* e da câmara decresce, passando para 3 fps. Em todos os outros cenários, estes dois valores permanecem constantes à volta das 15 fps.

Tabela 4 - Resultados do teste de carga

Caso de Teste	Número de polígonos	Rendering FPS	Tracking FPS	Camera FPS
Modelos Prof. Hugo (apenas área a estimular); textura verde	9.516	59 fps	15 fps	15 fps
Modelos Brainder (dois hemisférios) decimados; texturas verde e azul	57.908	53 fps	15 fps	15 fps
Modelos Prof. Hugo; texturas RGB	63.267	42 fps	15 fps	15 fps
Modelos Prof. Hugo; texturas escala de cinzentos	63.267	43 fps	15 fps	15 fps
Modelos Prof. Hugo; texturas escala de vermelhos	63.267	45 fps	15 fps	15 fps
Modelos Prof. Hugo; texturas escala de vermelhos com transparências	63.267	45 fps	15 fps	15 fps
Modelos Brainder (28 áreas da parcelização DK) não decimados; texturas RGB	219.479	38 fps	15 fps	15 fps
Modelos Brainder (dois hemisférios) não decimados; texturas verde e azul	579.114	6 fps	3 fps	3 fps

Feita esta análise, quis-se comparar os resultados obtidos com aqueles apresentados no sítio web do Metaio. Aí, a empresa apresenta também gráficos com o desempenho esperada em função do tipo de ficheiros usado para os modelos, do sistema operativo e do equipamento usado. Neste teste de carga, essa configuração corresponde a ficheiros OBJ para os modelos, Android para o sistema operativo e um Samsung Galaxy S4 como equipamento. Na Figura 5.25 e na Figura 5.26 apresentam-se os gráficos daí retirados, correspondentes a essa mesma configuração.

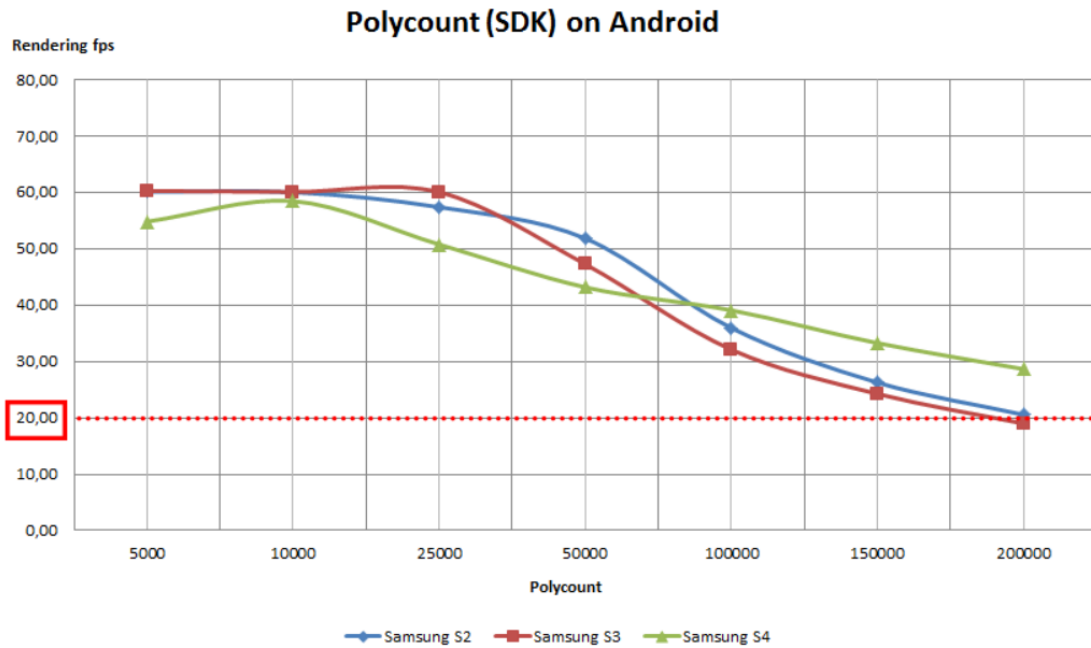


Figura 5.25 – Relação entre o número de polígonos a desenhar e o número de fps do rendering, relativamente a três smartphones Android diferentes [5]

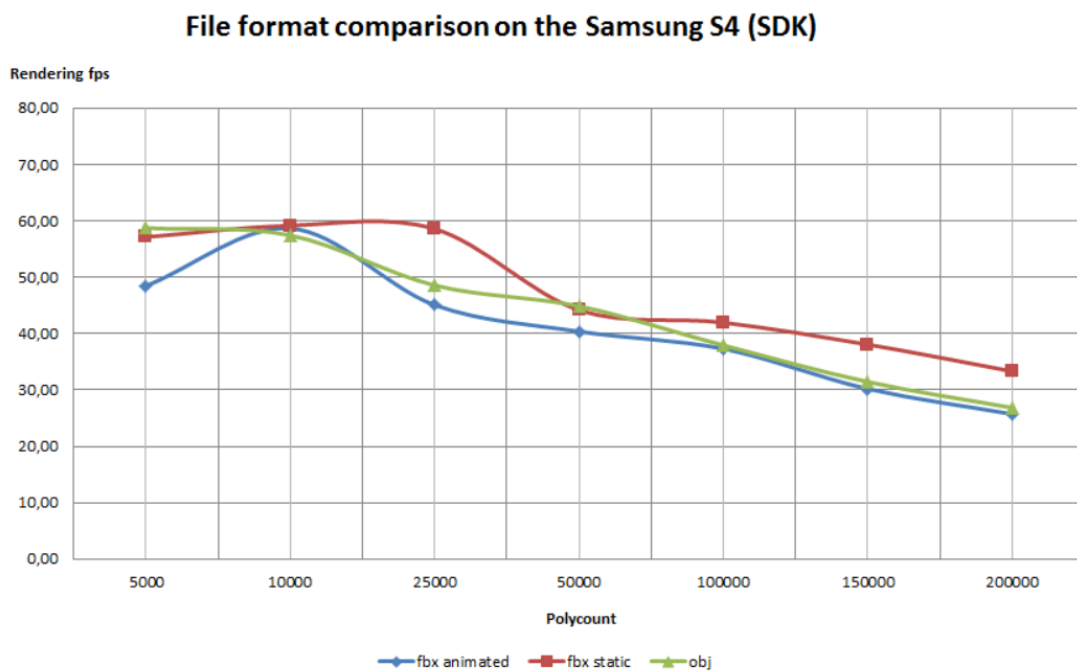


Figura 5.26 – Relação entre o número de polígonos a desenhar e o número de fps do rendering, relativamente a três formatos de ficheiro diferentes para os modelos, sempre num Samsung Galaxy S4 [5]

Nos gráficos apresentados verifica-se que os valores dados pelo Metaio não contradizem os valores observados no teste de carga. O número de polígonos poderia ainda ser mais reduzido para tentar aumentar o desempenho, ou então poder-se-ia mudar o formato de ficheiros para FBX (pois o Blender permite exportar modelos para esse

formato), mas tal mudança não se revelou necessária. Isto deve-se ao facto da aplicação funcionar de uma forma perfeitamente suave com o número de fps que apresenta atualmente, o que coincide com as informações no sítio web do Metaio, de que *rendering* de 25 fps é suficientes para a aplicação funcionar corretamente em tempo real. Apenas seriam necessárias mudanças caso o desempenho baixasse para além de 25 fps com os modelos que se estão a usar, ou caso se constate que 25 fps apesar de tudo não fornecem um desempenho suficientemente suave para os efeitos pretendidos.

Os modelos usados durante o resto do desenvolvimento do projeto são os modelos do projeto Brainder com a parcelização pelo atlas DK, decimados com razão de 0.1, o que equivale a um total de 63.262 polígonos para representar. Ao obter os modelos virtuais do cérebro de um paciente a partir de ressonância magnética pelo processo descrito na secção 5.7.1 cabe ao médico importá-los para o Blender e aplicar-lhes o modificador “*Decimate*”, podendo ajustar a razão do modificador para obter modelos finais com um maior ou menor número de polígonos, pelo que um bom desempenho da aplicação é sempre possível de conseguir.

5.7.5 Processamento dos modelos com o Blender Python

Como se explicou na secção anterior, os modelos virtuais obtidos a partir de ressonância magnética possuem demasiado detalhe, isto é, um número de polígonos demasiado grande para poderem ser desenhados pelo *smartphone* com a fluidez necessária. Como tal, é necessário serem importados para um programa de modelação 3D como o Blender, onde se lhes pode ser aplicado um modificador “*Decimate*” (no caso do Blender) que permite unir polígonos adjacentes para reduzir o número total de polígonos de cada modelo por uma razão determinada pelo utilizador.

Como já visto também, para que a aplicação cumpra os seus objetivos é necessários que o córtex cerebral seja representado não num só modelo mas em vários modelos correspondentes às diferentes áreas, de acordo com uma parcelização que divida o córtex num maior ou menor número de áreas. Assim, é natural que o modificador “*Decimate*” tenha de ser aplicado a um grande conjunto de modelos.

Infelizmente, porém, o Blender não permite importar mais que um modelo de cada vez, pelo que o processo de importar todos os modelos para o Blender teria de ser feito

importando os modelos um a um, o que se torna pouco prático quando existem muitos modelos para importar, como é o caso. Também a aplicação do modificador “Decimate” teria de ser feita para todos os modelos individualmente, apesar de ser possível aplicar o modificador a um e depois copiar o modificador com as mesmas definições para os restantes, seleccionando-os um a um. Para exportar cada modelo para um ficheiro diferente, como é necessário, também teria de se realizar o procedimento modelo a modelo, pelo que todo o trabalho de processamento dos modelos no Blender seria bastante demorado e pouco prático, para além de exigir ao médico conhecimentos sobre o Blender.

Para resolver este problema, decidiu-se criar um script para autonomizar o processo, fazendo uso da integração da linguagem de programação Python no ambiente Blender. Esta integração consiste na possibilidade de executar comandos do Blender através de instruções Python, graças a um módulo Python criado pelos desenvolvedores do Blender e que pode ser incluído nos scripts para dar acesso a dados, classes e funções do Blender. Os scripts podem ser criados diretamente no Blender ou importados a partir de ficheiros já criados, e o Blender possui um interpretador Python integrado que permite correr os scripts dentro do programa, visualizando logo de seguida as alterações à cena 3D que se está a trabalhar [49].

O script criado utiliza o ficheiro de definição das geometrias (ver secção 5.7.3) para ler os nomes de cada um dos ficheiros que contém os modelos, um a um. Para cada ficheiro, o script importa o modelo respetivo para o Blender, aplica-lhe o modificador “Decimate” com razão de 0.1, exporta o modelo com o mesmo nome para uma nova pasta e apaga por fim o modelo da cena 3D, antes de passar ao modelo seguinte.

Com a utilização deste script o processo de tratamento dos modelos no Blender passa a demorar alguns segundos apenas, sendo apenas necessário que o ficheiro de definição das geometrias já tenha sido criado. Em caso de necessidade de outras edições aos modelos no Blender, adaptações ao script podem ser feitas para incluir essas edições. O script criado pode ser encontrado no Anexo G.

5.8 Introdução de informação na aplicação

Ao longo das secções anteriores analisou-se o funcionamento da aplicação nas suas várias vertentes. Percebeu-se como funcionava o *tracking*, componente essencial para o

modo de Realidade Aumentada, sendo que a aplicação precisa de ler um ficheiro XML de configuração quando é iniciada (ver secção 5.3.2). Viu-se ainda a importância de ter cada área do cérebro como um modelo virtual separado (guardado num ficheiro individual), bem como a necessidade de indicar, para cada uma dessas áreas, qual a cor da textura que lhe deve ser aplicada, bem como a que lobo e hemisfério essa área pertence, para permitir a visualização seletiva por lobos ou hemisférios. Viu-se que a resposta a estes últimos problemas era dada através da leitura pela aplicação de um ficheiro de texto de definição das geometrias (ver secção 5.7.3).

Assim, existem alguns ficheiros externos que a aplicação precisa de ler e que têm de ser criados/atualizados e gravados no *smartphone* pelo utilizador: o ficheiro XML de configuração do *tracking*, o ficheiro TXT de definição das geometrias e ainda os vários ficheiros OBJ contendo os modelos das várias áreas em que o cérebro deve estar particionado. É importante perceber então onde se devem gravar esses ficheiros e como é que a aplicação os consegue ler.

Para simplificar este problema, a aplicação procura os ficheiros num diretório específico da memória do *smartphone*, reconhecendo-os também a partir do nome. Assim, deve ser criada (se ainda não existir) uma pasta de nome “BrainARVR” dentro do diretório raiz do armazenamento do dispositivo móvel (o mesmo diretório onde por predefinição se encontram as pastas “Alarms”, “DCIM”, “Downloads”, “Movies”, “Music”, “Notifications”, “Pictures”, “Podcasts” e “Ringtones”, criadas pelo sistema operativo). É dentro desta pasta que se devem colocar todos os ficheiros que a aplicação deve ler, como os ficheiros de configuração do *tracking*, os ficheiros de definição das geometrias e os ficheiros com os modelos virtuais.

Ao ficheiro de configuração do *tracking* deve ser dado o nome “trackingConfig.xml”, para que a aplicação o consiga reconhecer como o ficheiro de configuração a carregar. Pelo mesmo motivo, o ficheiro de definição das geometrias deve ser chamado “geometryDef.txt”. No que toca aos modelos virtuais, os seus ficheiros não precisam de ter nomes específicos, porque não é assim que a aplicação os encontrará. Em lugar disso, as localizações dos modelos são dadas à aplicação a partir da leitura do ficheiro de definição das geometrias. Os caminhos indicados para os modelos devem ser dados relativamente à pasta “BrainARVR”, ou seja, um modelo de nome

“model_temporal_pole.obj” que esteja guardado dentro de uma pasta “ModelsPatientA” dentro da pasta “BrainARVR” deve ser indicado com o caminho “ModelsPatientA/model_temporal_pole.obj”.

Dentro da pasta “BrainARVR” o utilizador pode guardar mais ficheiros do que os que vão ser lidos pela aplicação. Por exemplo, o utilizador pode colocar na pasta ficheiros de configuração do *tracking* e de definição das geometrias de vários pacientes. Para visualizar dados de outro paciente bastará mudar o nome dos ficheiros correspondentes a esse paciente para o nome reconhecido pela aplicação, “trackingConfig.xml” ou “geometryDef.txt”, conforme o tipo de ficheiro. Os ficheiros relativos ao paciente antigo também têm primeiro de ter o seu nome mudado, ou então devem ser colocados noutra pasta, para não interferirem com os ficheiros novos que se querem ler.

Em relação aos modelos, podem ser guardados em pastas conjuntos de modelos relativos a vários pacientes, ou mesmo conjuntos de modelos relativos ao mesmo paciente segundo parcelizações diferentes. Para cada conjunto de modelos deve existir um ficheiro de definição das geometrias, que indique o caminho para esses modelos e associe um lobo, um hemisfério e uma cor a cada área.

Como trabalho futuro, numa possível versão mais avançada da aplicação, poderá ser criada uma interface de utilizador para se escolherem, ao iniciar a aplicação, quais os ficheiros a carregar. A aplicação poderá ainda manter um registo dos pacientes e dos modelos e ficheiros correspondentes, sendo apenas necessário antes de cada utilização escolher de que paciente se devem carregar os dados ou, caso seja a primeira utilização nesse paciente, introduzir os seus dados, incluindo a localização dos dois ficheiros.

5.9 Conclusão

Neste capítulo descreveu-se a aplicação criada, *Brain AR/VR*, a sua arquitetura, o funcionamento de cada um dos modos, de RA e de RV, e as funcionalidades implementadas. Detalharam-se ainda os desafios ligados ao desenvolvimento de cada componente da solução, em particular: *tracking*, visualização, perceção de profundidade e estereoscopia, aquisição e processamento dos modelos e introdução prévia da informação na aplicação. O capítulo seguinte é relativo à avaliação da aplicação através de testes de utilizador e à análise dos resultados obtidos.

Capítulo 6

Avaliação da aplicação

Neste capítulo apresenta-se o procedimento de avaliação da aplicação através de testes com utilizadores e os resultados obtidos, fazendo-se uma análise dos mesmos.

6.1 Metodologia

Na fase final do projeto procedeu-se à realização de testes com utilizadores, seguindo um modelo de entrevista por inquirição contextual. Os grandes objetivos destes testes foram uma avaliação da qualidade da aplicação *Brain AR/VR* e a recolha de sugestões para melhorias futuras e adição de novas funcionalidades. Foi criado um guião de entrevista para os testes (Anexo H), no qual se estabeleceu uma sequência de tarefas para o utilizador realizar com a aplicação juntamente com perguntas para obter a avaliação e sugestões do utilizador à medida que executava as tarefas. O utilizador foi guiado por um interlocutor, que lia o guião passo a passo, dava as indicações ao utilizador, fazia as perguntas e registava as respostas no guião.

O guião continha uma secção inicial para recolher dados pessoais do utilizador e uma autoavaliação dos seus conhecimentos e experiência com *smartphones*, ecrãs táteis e Realidade Aumentada ou Virtual. Em seguida, fornecia-se ao utilizador o *smartphone* de teste com a aplicação e dava-se início à leitura do guião com as tarefas e as perguntas. A avaliação efetuada pode dividir-se nos seguintes componentes:

- Modo de RA
 - Processo de calibração das marcas
 - Alinhamento dos modelos
 - Glifos do aparelho de estimulação
- Modo de RV
- Visualização Seletiva
- Apreciações globais

Em relação ao formato das perguntas no guião, estas podem dividir-se essencialmente em dois tipos:

- Pergunta de classificação – pede-se ao utilizador que aprecie um certo atributo da aplicação, segundo uma escala de Likert de mínimo 1 e máximo 5;
- Pergunta de resposta aberta – pede-se ao utilizador que dê a sua opinião sobre um determinado aspeto da aplicação, referindo os pontos negativos ou positivos que encontrou, justificando as suas apreciações, ou dando sugestões de mudanças ou melhorias futuras.

Para além destes dois tipos de perguntas, existiam ao longo do guião alguns espaços a preencher autonomamente pelo interlocutor do utilizador, registando aspetos do decorrer do teste que interessavam analisar. Estes espaços consistiam em perguntas de escolha múltipla e espaços para registar uma resposta curta, em texto ou um valor numérico ou de tempo.

Cada teste teve uma duração a variar entre 20 e 50 minutos, em função da quantidade de *feedback* dada por cada utilizador. Os testes foram efetuados com o *smartphone* usado durante o desenvolvimento da aplicação e com a touca com marcas colocada no modelo de cabeça feminina usado. Uma régua com uma marca foi usada para simular o aparelho de estimulação. Todos os testes foram realizados dentro de portas, em gabinetes médicos ou salas do IBEB, o que permitiu executar os testes em ambientes semelhantes àqueles em que a aplicação pode ser usada e em que a luminosidade não impossibilitasse a deteção das marcas. No Anexo I podem encontrar-se algumas fotografias dos testes de utilizador realizados.

6.2 Perfil dos Utilizadores

Os testes foram realizados com 11 utilizadores voluntários: médicos, docentes, investigadores e/ou alunos da FCUL. As idades estão compreendidas entre os 22 e os 55 anos, 8 utilizadores são do sexo masculino e 3 são do sexo feminino. Dois dos utilizadores são médicos que trabalham e têm experiência na área da Neurologia, um na área da Neuropsiquiatria e o segundo nas áreas da Neurologia e Neurofisiologia, sendo um dos

maiores especialistas do país em EMT. Outra percentagem dos utilizadores corresponde a professores e investigadores que trabalham no IBEB, maioritariamente na área da Física e da Medicina, com conhecimentos sobre EMT. Os restantes utilizadores correspondem a estudantes de mestrado na área da Engenharia Biomédica, todos trabalhando no IBEB.

Foi pedido aos utilizadores, no início de cada teste, que se autoavaliassem (numa escala em que 1 é o mínimo e 5 é o máximo) em relação a três pontos diferentes: experiência de utilização de *smartphones*; experiência de utilização de ecrãs táteis; conhecimentos de Realidade Aumentada ou Virtual. Na Figura 6.1 pode observar-se a distribuição das respostas dos utilizadores. A mediana de cada distribuição é apresentada entre parênteses abaixo de cada gráfico.

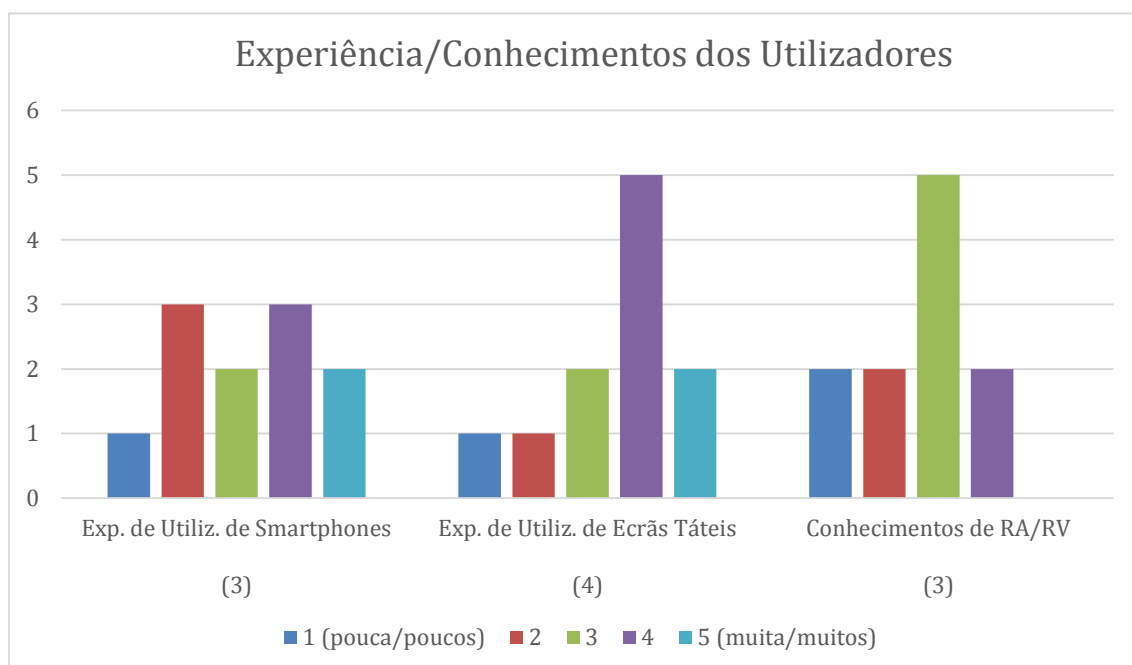


Figura 6.1 – Gráficos da distribuição dos utilizadores de teste por níveis autoatribuídos numa escala de 1 a 5 em relação a: experiência de utilização de smartphones, experiência de utilização de ecrãs táteis e conhecimentos de Realidade Aumentada ou Virtual

6.3 Análise dos Resultados

São de seguida apresentados os resultados obtidos pelos testes de utilizador, com a exposição das respostas às principais perguntas e as críticas e sugestões mais relevantes. As perguntas serão identificadas com o seu número no guião (ver Anexo H). Para cada gráfico apresentado da distribuição das respostas dos utilizadores, a mediana dessa distribuição é indicada abaixo do mesmo, entre parênteses.

6.3.1 Modo de RA

A avaliação do modo de RA dividiu-se em três componentes principais: o processo de calibração, o alinhamento dos modelos e os glifos do aparelho de estimulação.

Começou-se por se explicar ao utilizador como calibrar as marcas e pedir-lhe que efetuasse o processo, tendo depois sido registadas as apreciações do utilizador sobre o mesmo. Em seguida, pediu-se ao utilizador que rodasse em torno do modelo de cabeça feminina para observar o modelo de vários ângulos. Pediu-se então ao utilizador que avaliasse o alinhamento do modelo e desse sugestões relativas à representação/visualização do modelo do cérebro. Depois deu-se a régua com a marca ao utilizador para que este simulasse a estimulação de uma área do cérebro representada a vermelho, de modo a que este observasse o glifo do aparelho de estimulação e o código de cores associado. Avaliou-se então a perceção de profundidade, a utilidade e intuitividade do código de cores, a representação visual do aparelho e a experiência de estimulação. Guiou-se então o utilizador para o menu de opções, onde este alterou o glifo usado para representar o aparelho de estimulação e em seguida avaliou-se novamente a perceção de profundidade, a representação visual do aparelho e a experiência de estimulação. Por fim, pediu-se uma apreciação global do modo de RA.

Na Figura 6.2 pode observar-se a distribuição das respostas dos utilizadores em relação à facilidade do processo de calibração (2a), à duração do processo de calibração (2b) e à utilidade dos indicadores que aparecem no ecrã para ajudar no processo (2c). Todos os utilizadores consideraram o processo de calibração fácil e rápido. Em relação aos indicadores, a grande maioria dos utilizadores considerou a sua utilidade bastante elevada, tendo um utilizador apenas feito uma apreciação negativa por considerar que os indicadores não deveriam ser textuais mas sim algo mais gráfico. Também muitos dos restantes utilizadores sugeriram a utilização de um esquema gráfico em lugar dos indicadores textuais, enquanto outros sugeriram mudanças de cor/tamanho nos indicadores atuais e outros sugeriram a adição de um sinal sonoro quando uma marca era calibrada.

Em relação ao processo de calibração em si, alguns utilizadores referiram que a luminosidade causou algumas dificuldades ao processo e a importância da existência de um manual de utilizador da aplicação, onde os problemas que podem ocorrer na deteção

das marcas sejam mencionados. Um utilizador sugeriu ainda que se identificassem as marcas não pelo seu número mas sim pela sua posição na cabeça do paciente, através da nomenclatura utilizada em eletroencefalogramas (EEGs).

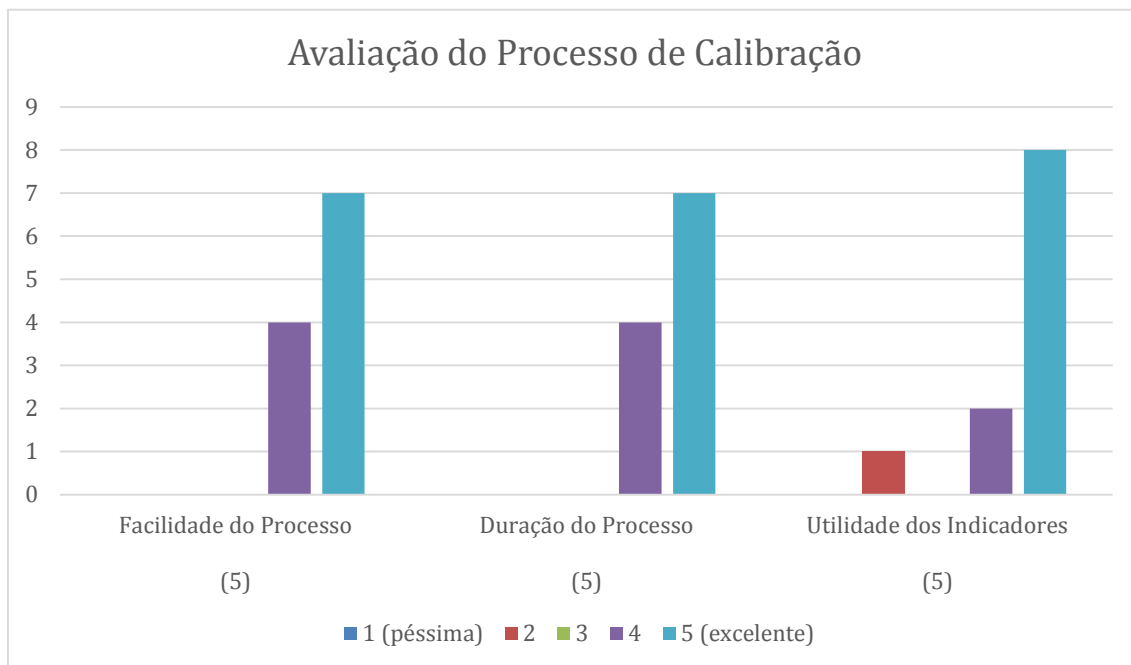


Figura 6.2 - Gráficos da distribuição das respostas relativamente ao processo de calibração em três componentes: facilidade, duração e utilidade dos indicadores

Na Figura 6.3 pode observar-se a distribuição das respostas dos utilizadores em relação ao alinhamento do modelo virtual no modo de RA (3c). A apreciação é positiva, mas vários utilizadores indicaram, enquanto observações/sugestões, que era preciso melhorar a estabilidade do modelo e, em alguns casos, corrigir o seu alinhamento.

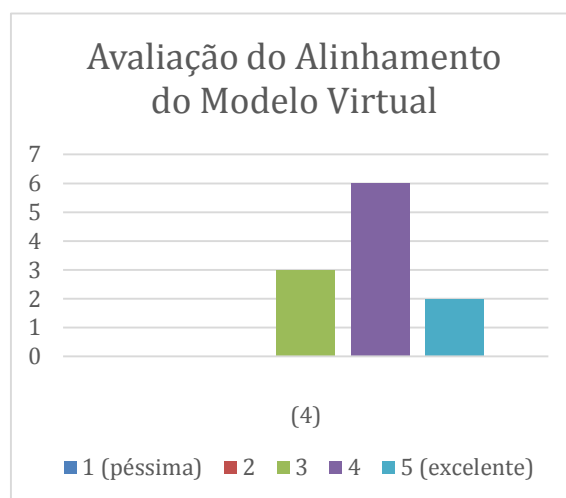


Figura 6.3 – Gráfico da distribuição das respostas relativamente ao alinhamento do modelo virtual no modo de RA

No que diz respeito à representação/visualização do modelo do cérebro, vários utilizadores consideraram que o modelo devia ser mais realista – faltando sombras no modelo para que se percebessem os contornos e relevos do modelo e se melhorasse a percepção tridimensional – apesar de compreenderem que esse realismo introduziria uma maior necessidade de computação ao *hardware*. Outros utilizadores referiram ainda que as áreas deveriam estar coloridas com mais cores diferentes, para que se distingam melhor entre si e um utilizador sugeriu que a área a estimular fosse representada a azul e não a vermelho.

Na Figura 6.4 pode observar-se a distribuição das respostas dos utilizadores em relação à utilidade da indicação colorida da distância no glifo do aparelho de estimulação (4c) e à intuitividade do código de cores utilizado para essa indicação (4d). Os resultados são positivos, apesar de alguns utilizadores experientes em EMT terem considerado que a indicação colorida da distância não é necessária, uma vez que o médico tem sempre de aplicar a técnica colocando o aparelho de estimulação em contato com a cabeça do paciente.

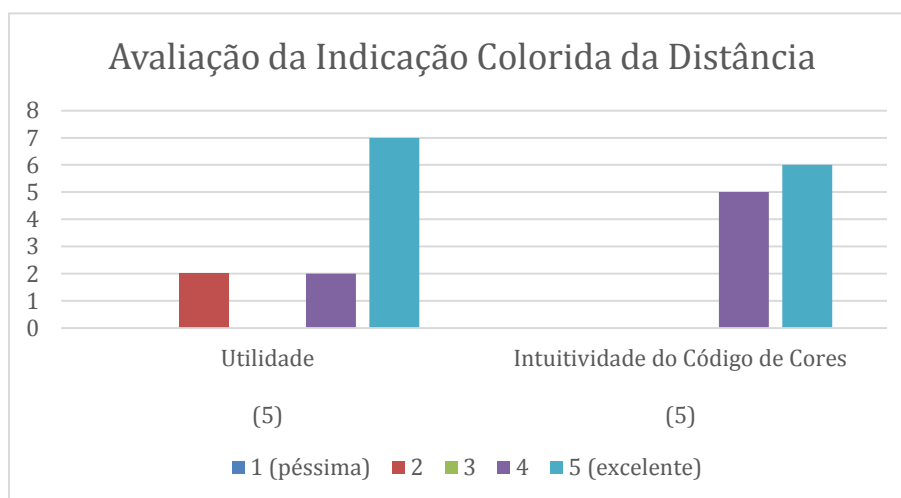


Figura 6.4 - Gráficos da distribuição das respostas relativamente à utilidade da indicação colorida da distância e intuitividade do código de cores usado

Cada utilizador avaliou três dos aspetos do modo de RA duas vezes, uma usando o glifo “Arrow” para representar o aparelho de estimulação e outra usando o glifo “Coil”: a percepção de profundidade (4b e 5a), a representação visual do aparelho de estimulação (4e e 5b) e a qualidade da experiência de estimulação (4f e 5c). Na Figura 6.5 podem observar-se as distribuições das respostas dos utilizadores em relação a estes três aspetos tanto para um glifo como para o outro. De um modo geral, os resultados foram melhores

com o glifo “Coil”, que se assemelha mais a um aparelho de estimulação real. Apesar da representação do aparelho mais bem avaliada ter sido a do glifo “Arrow”, o glifo “Coil” foi avaliado pelos utilizadores como fornecendo uma muito melhor percepção de profundidade na visualização e, conseqüentemente, uma melhor experiência de aplicação da técnica no geral. A melhoria na percepção de profundidade, segundo um utilizador, deve-se ao toro usado no glifo “Coil”, que evidencia a orientação em que o aparelho de estimulação está a ser usado e as rotações que se lhe apliquem, algo que com o glifo “Arrow” é mais difícil de perceber.

Um dos utilizadores mais experientes em EMT fez uma análise interessante dos glifos usados, associando cada um deles a um dos diferentes formatos que os aparelhos de estimulação têm, tendo em conta a forma como a intensidade do campo magnético está distribuída em aparelhos de cada um desses formatos. A intensidade do campo magnético é maior na zona do aparelho onde está a bobina por onde a corrente elétrica passa. Além disso, é importante para o médico visualizar o sentido em que a corrente elétrica passa na bobina.

O glifo “Arrow” estaria então associado a um aparelho de estimulação/bobina em forma de oito, pois a corrente elétrica passa duas vezes no centro do aparelho de estimulação, sempre no mesmo sentido, logo nesse local e nesse sentido é onde o campo magnético é mais forte. A seta do glifo “Arrow” é indicada, então, na perspectiva do utilizador, para representar bobinas em forma de oito.

Quanto ao glifo “Coil”, este representaria bem apenas bobinas circulares, nas quais a sua representação foi de facto baseada. Porém, com bobinas deste formato o médico costuma utilizar os dois sentidos da corrente elétrica, consoante a necessidade, e para tal muitas vezes vira o aparelho de estimulação ao contrário, o que tornaria a marca invisível, pelo seria necessário colocar uma marca de cada lado do aparelho. Além disso, seria importante a visualização do glifo incluir as setas com o sentido da corrente elétrica, que teriam de estar posicionadas exatamente nos locais onde o campo magnético é mais forte, locais esses que variam com o tamanho de cada bobina. O mesmo utilizador sugeriu ainda, em relação ao glifo “Coil”, que se aumentasse o diâmetro interior do toro, diminuindo o buraco central, e que se colocasse transparência no glifo para que o cérebro

continuasse visível por baixo de uma representação tão grande do aparelho de estimulação.

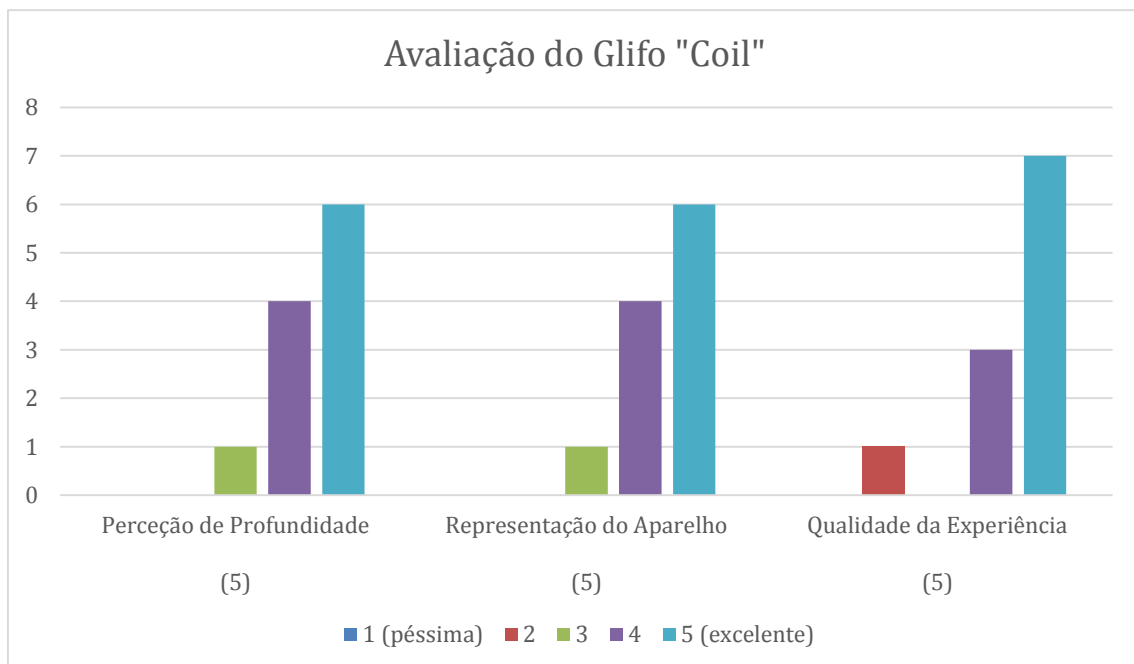
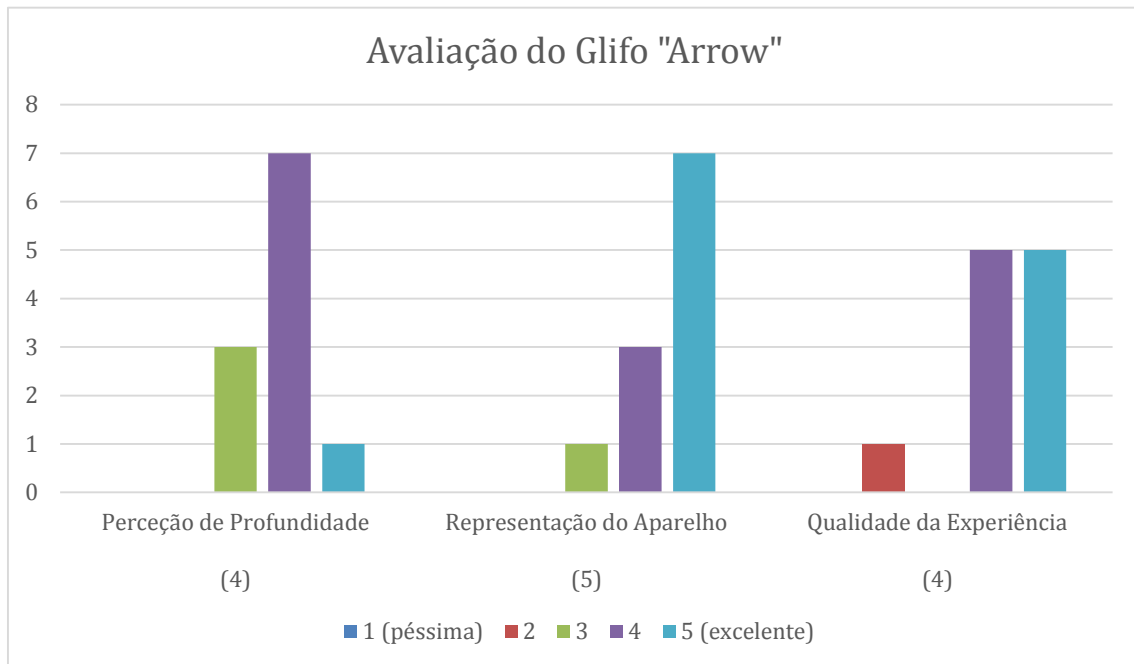


Figura 6.5 – Comparação entre os gráficos da distribuição das respostas relativamente à perceção de profundidade, representação visual do aparelho de estimulação e a qualidade da experiência de estimulação para os dois glifos alternativos do aparelho de estimulação: "Arrow" e "Coil"

Na Figura 6.6 pode observar-se a distribuição das respostas dos utilizadores em relação a uma apreciação global do modo de RA (5d). No geral, os resultados da avaliação deste modo foram positivos, tendo sido recolhidas sugestões bastante úteis para

melhoramentos futuros. O processo de calibração foi bem avaliado pelos utilizadores, mas é preciso melhorar o alinhamento dos modelos e a sua estabilidade, problemas que decorrem de falhas na deteção das marcas. O realismo do modelo do cérebro também pode ser melhorado e, em relação aos glifos do aparelho de estimulação, a indicação colorida da distância não é uma função essencial. O glifo “Coil” proporciona uma melhor perceção de profundidade que o glifo “Arrow”, mas ambos podem ser melhorados e usados para representar aparelhos de estimulação de diferentes formatos, existindo espaço para se criar uma visualização mais realista e informativa para o médico.

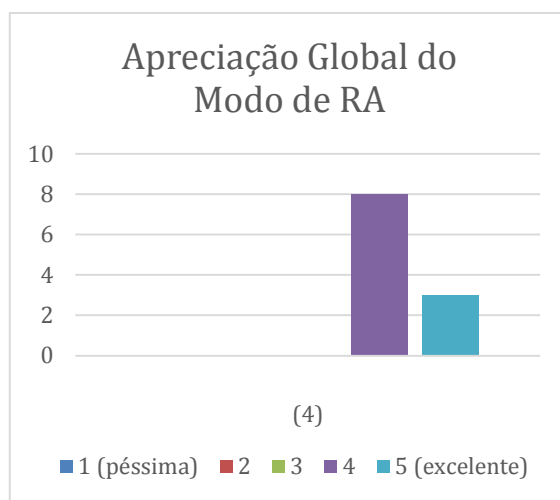


Figura 6.6 – Gráfico da distribuição das respostas relativamente à apreciação global do modo de RA

6.3.2 Modo de RV

A avaliação do modo de RV incidiu sobre os três tipos de interações básicas que se podem efetuar nesse modo: *zoom-in/out* do modelo, rotação do mesmo e a utilização das vistas predefinidas. Pediu-se ao utilizador que ajustasse o tamanho do modelo, que o rodasse e depois que alterasse o eixo de rotação e voltasse a rodar. Em seguida, pediu-se que o utilizador pensasse numa orientação a partir da qual gostava de ver o modelo e tentasse colocá-lo nessa orientação rodando o modelo. Pediu-se ainda que o utilizador experimentasse as vistas predefinidas e fez-se depois a avaliação.

Os aspetos avaliados foram: a facilidade de ajustar o tamanho do modelo (6a), a intuitividade do gesto para ajustar o tamanho do modelo (6b), a facilidade de colocar o modelo numa orientação desejada sem recorrer às vistas predefinidas (6c), a intuitividade do gesto para rodar o modelo (6d) e a utilidade das vistas predefinidas (6e). Na Figura 6.7

pode observar-se a distribuição das respostas dos utilizadores em relação a estes cinco aspetos.

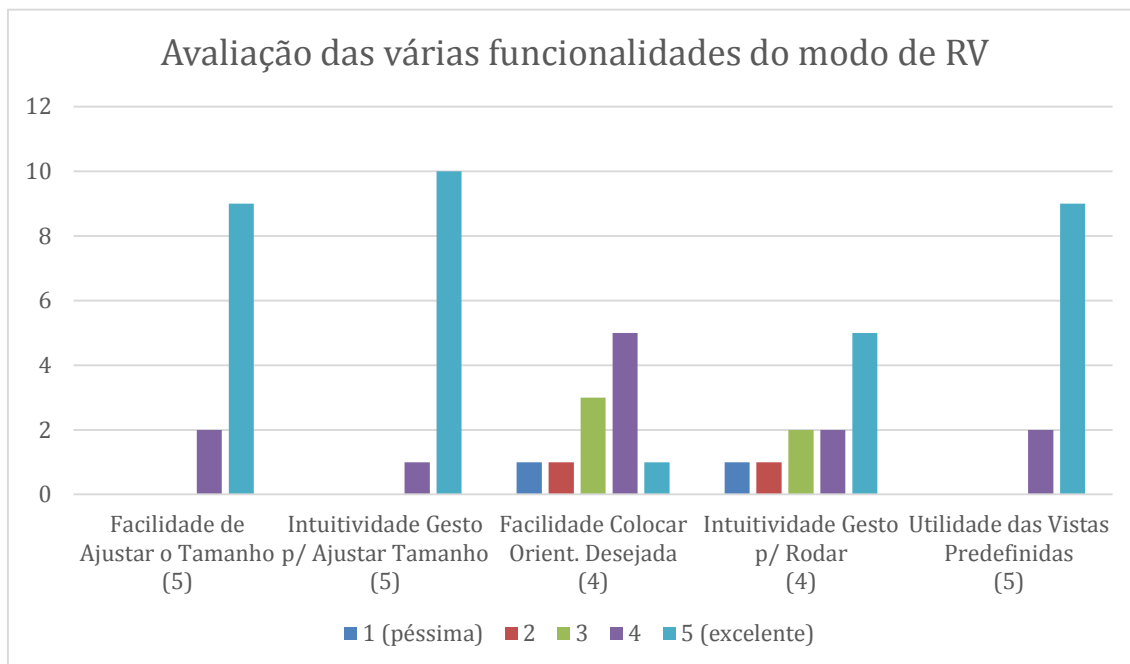


Figura 6.7 – Gráficos da distribuição das respostas relativamente à facilidade de ajustar o tamanho do modelo, intuitividade do gesto para ajustar o tamanho, facilidade de colocar o modelo numa orientação desejada sem recorrer às vistas predefinidas, intuitividade do gesto para rodar o modelo e utilidade das vistas predefinidas

A avaliação dos utilizadores foi extremamente positiva no que diz respeito ao *zoom-in/out* do modelo e às vistas predefinidas. Em relação à rotação do modelo, porém, os resultados não foram tão bons, apesar de continuarem positivos. Este facto foi reforçado pelas observações/sugestões dos utilizadores, que deram algumas ideias alternativas sobre como gostariam que se controlasse a rotação do modelo. Em relação aos botões de mudança do eixo de rotação, foi desejado que existisse um realce para se saber qual dos eixos estava ativo em cada momento e também que se aumentassem o tamanho do texto desses botões. Foi ainda sugerido por alguns utilizadores que fosse possível transladar o modelo ou pelo menos que a sua posição estivesse mais acima no ecrã, para que os botões da interface não o ocultassem em alguns momentos. No que diz respeito às vistas predefinidas, dois utilizadores sugeriram que se alterasse nos botões o nome de cada uma das vistas segundo a nomenclatura usada por neurologistas: “Anterior”, “Posterior”, “Left”, “Right”, “Superior” e “Inferior” em vez, respetivamente, de “Front”, “Back”, “Left”, “Right”, “Top” e “Bottom”. Um utilizador sugeriu ainda que fosse possível configurar na aplicação o tamanho/orientação inicial do modelo, em vez da vista frontal predefinida.

Na Figura 6.8 pode observar-se a distribuição das respostas dos utilizadores em relação a uma apreciação global do modo de RV (6f). No geral, os resultados da avaliação deste modo foram positivos, tendo sido recolhidas sugestões bastante úteis para melhoramentos futuros. Em particular, a alteração mais importante será na interação de rotação do modelo, mas também em alguns aspetos da interface. É útil referir que a avaliação efetuada no modo de RA à representação do modelo do cérebro pode ser transportada para este modo, sendo que mudanças no realismo do modelo, por exemplo, afetariam os dois modos.

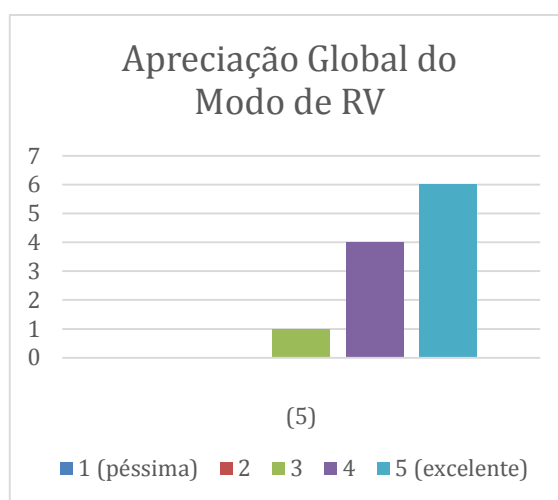


Figura 6.8 - Gráfico da distribuição das respostas relativamente à apreciação global do modo de RV

6.3.3 Visualização Seletiva

Para além dos dois modos de uso da aplicação, avaliou-se também a funcionalidade de Visualização Seletiva (que pode ser usada com ambos) e com ela o menu de opções. Pediu-se então ao utilizador que entrasse no menu de opções, observasse como todas as caixas de verificação estavam marcadas, desmarcasse dois lobos e depois observasse como tinham ficado automaticamente desmarcadas as caixas de verificação relativas às áreas do cérebro pertencentes a esses dois lobos e ainda que confirmasse a seleção e observasse que no modo de RV apenas eram visíveis as áreas dos outros dois lobos do cérebro. Pediu-se em seguida ao utilizador que observasse que o mesmo acontecia no modo de RA e depois que voltasse ao menu de opções para desmarcar mais algumas áreas e confirmar o resultado.

A avaliação efetuada em seguida incidiu sobre três aspetos: organização do menu de opções (7a), facilidade de alterar as áreas visíveis (7b) e utilidade da funcionalidade

de Visualização Seletiva (7c). Na Figura 6.9 pode observar-se a distribuição das respostas dos utilizadores em relação a estes três aspetos.

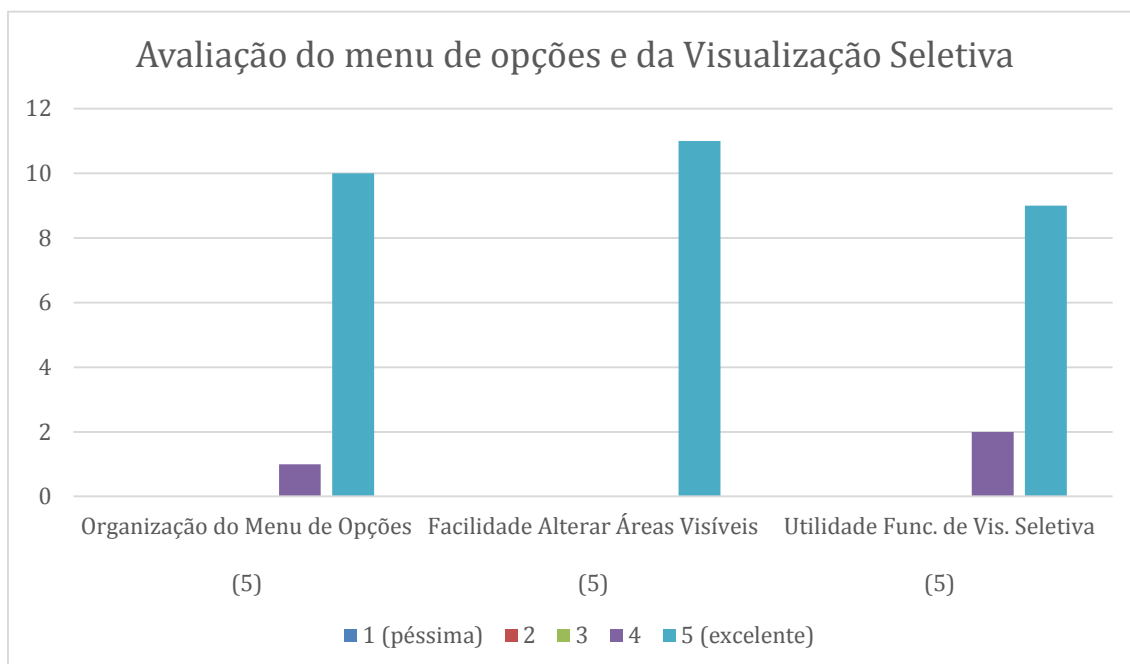


Figura 6.9 - Gráficos da distribuição das respostas relativamente à organização do menu de opções, facilidade de alterar as áreas visíveis e utilidade da funcionalidade de Visualização Seletiva

Os resultados são extremamente positivos para os três aspetos avaliados, tendo os utilizadores aprovado a organização do menu de opções e apreciado bastante a funcionalidade de Visualização Seletiva, que consideraram bastante útil. As sugestões dadas por alguns utilizadores consistiram em incluir uma caixa de verificação no menu de opções para marcar/desmarcar automaticamente todas as áreas do cérebro e também em visualizar as áreas desmarcadas com transparência em vez de completamente invisíveis, para que se continue a ter uma ideia da estrutura global do cérebro. Um utilizador referiu ainda a pertinência da parcelização do cérebro a usar, considerando que é um detalhe muito importante que deve ser bem estudado.

Na Figura 6.10 pode observar-se a distribuição das respostas dos utilizadores em relação a uma apreciação global da funcionalidade de Visualização Seletiva (7d). No geral, os resultados da avaliação desta funcionalidade foram extremamente positivos, tendo sido recolhidas algumas sugestões bastante úteis para melhoramentos futuros. Os utilizadores apreciaram bastante a flexibilidade de visualização oferecida e consideraram existir muito potencial para se incluírem outras opções de personalização da visualização.

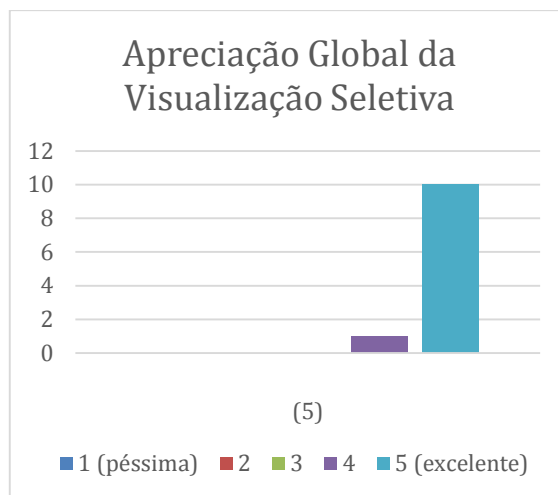


Figura 6.10 – Gráfico da distribuição das respostas relativamente à apreciação global da funcionalidade de Visualização Seletiva

6.3.4 Apreciações Finais

Depois da avaliação dos dois modos de uso e da funcionalidade de Visualização Seletiva, pediram-se umas últimas tarefas e apreciações ao utilizador. Primeiro, foi-lhe pedido que terminasse a aplicação. O objetivo era registar a forma como a maioria dos utilizadores o fazia, analisando se conseguiam detetar o botão “X” no canto superior direito e perceber que era lá que deviam carregar (8a). A grande maioria dos utilizadores (8 em 11) utilizaram esse botão “X” para terminar a aplicação, validando a sua visibilidade e intuitividade. Em seguida, pediu-se ao utilizador que voltasse a entrar na aplicação e a calibrar as marcas, para se medir o tempo que o utilizador demorava no processo de calibração (9a). Em média, a duração do processo foi de aproximadamente 1 minuto e 5 segundos. Mediu-se ainda o tempo que o utilizador demorava a utilizar a funcionalidade de Visualização Seletiva para conseguir visualizar e começar a simular a estimulação sobre uma área mais interior do cérebro assinalada a verde (9b). Em média, o utilizador demorou 36 segundos até começar a simulação.

Após as medições de tempos, pediu-se ao utilizador algumas apreciações globais e observações/sugestões finais sobre alguns aspetos da aplicação: as funcionalidades (9c), as capacidades de visualização fornecidas (9e), a interface de utilizador (9g) e a utilidade/relevância da aplicação (9i). Na Figura 6.11 pode observar-se a distribuição das respostas dos utilizadores em relação às apreciações globais destes quatro aspetos.

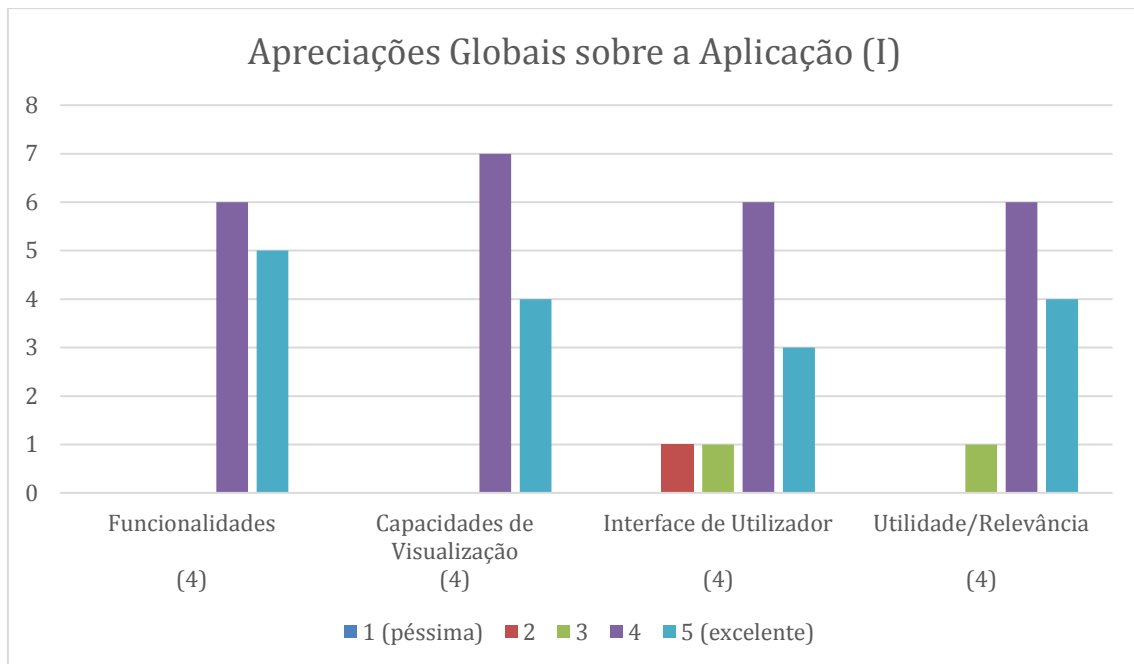


Figura 6.11 – Gráficos da distribuição das respostas relativamente às apreciações globais das funcionalidades da aplicação, capacidades de visualização fornecidas, interface de utilizador e utilidade/relevância da aplicação

Os resultados obtidos são positivos, sendo as funcionalidades da aplicação bastante bem apreciadas pelos utilizadores, seguidas das capacidades de visualização fornecidas, da utilidade/relevância da aplicação e, por fim, da interface de utilizador. Em termos de funcionalidades, os utilizadores realçaram a necessidade de melhorar a estabilidade e o alinhamento do modelo do cérebro. Sugeriram também a adição de novas funcionalidades à aplicação, como por exemplo visualizar as posições de elétrodos intracranianos, e que o cérebro fosse parcelizado de acordo com mapas funcionais. Foi ainda referido que seria bastante útil que a aplicação funcionasse em óculos de RA, para libertar a mão do utilizador de segurar o *smartphone*. Relativamente às capacidades de visualização da aplicação, alguns utilizadores realçaram a necessidade de mais realismo nos modelos. No que toca à interface, foi sugerida a existência de um botão e ecrã de ajuda. Também foi sugerido que se pudesse alterar o idioma da aplicação, pelo menos entre inglês e português.

Na Figura 6.12 pode rever-se a distribuição das respostas dos utilizadores em relação às apreciações globais do modo de RA, modo de RV e funcionalidade de Visualização Seletiva. Esta última funcionalidade foi extremamente bem apreciada pelos utilizadores, enquanto o modo de RA ficou com a apreciação mais baixa das três, o que se justifica pelos problemas ainda existentes na estabilidade e alinhamento do modelo virtual do cérebro, devidos às limitações na capacidade de *tracking* das marcas.

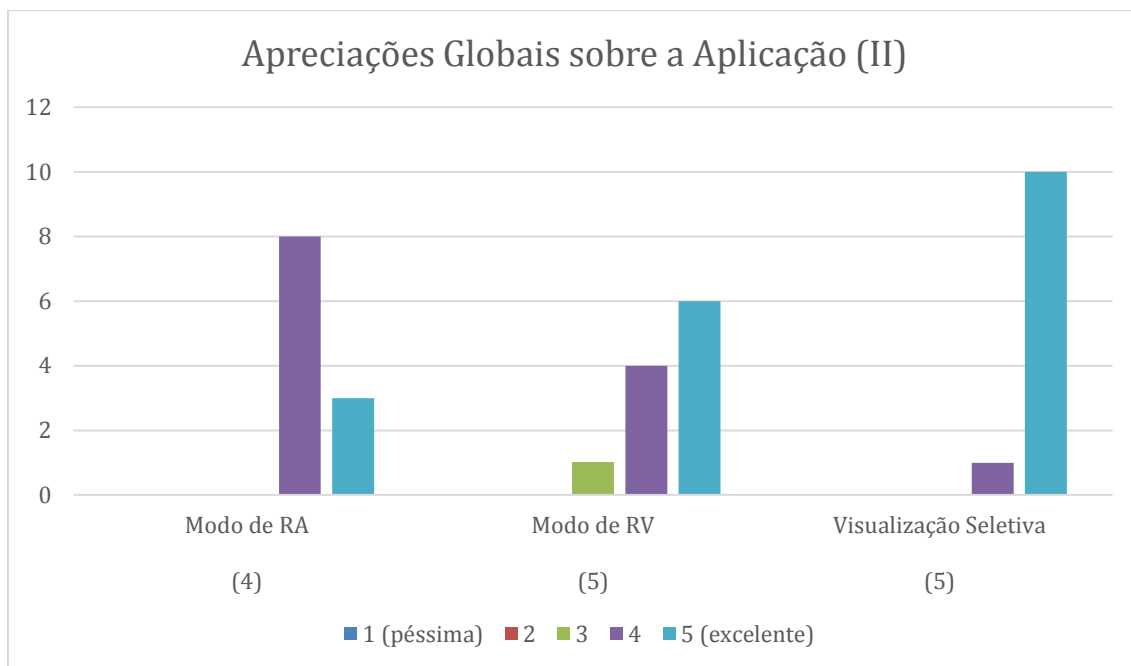


Figura 6.12 – Gráficos da distribuição das respostas relativamente às apreciações globais do modo de RA, modo de RV e funcionalidade de Visualização Seletiva

Concluindo, a aplicação foi bem recebida pelos utilizadores, que se mostraram interessados nas suas funcionalidades, gostaram da versão que testaram e deram sugestões bastante úteis para melhoramentos futuros. Os resultados mais importantes desta avaliação são discutidos mais detalhadamente no capítulo seguinte.

Capítulo 7

Conclusões e Trabalho Futuro

Neste documento apresentou-se uma solução de Realidade Aumentada e Virtual para visualização de modelos virtuais do cérebro num *smartphone*. Foi criada uma aplicação Android – *Brain AR/VR* – com dois modos de uso: um de RA, em que o modelo virtual aparece sobre a imagem da câmara alinhado com a cabeça de um indivíduo, graças a uma touca com marcas que este tenha colocada, e outro de RV, em que apenas se vê o modelo virtual e é possível interagir-se com o mesmo através de toque no ecrã do dispositivo móvel. Incluiu-se também na aplicação uma funcionalidade de Visualização Seletiva das áreas do cérebro a visualizar, graças a uma parcelização do modelo que tem de ser efetuada previamente à inserção do modelo na aplicação.

A aplicação foi pensada, no seu modo de RA, para auxílio a médicos na execução da técnica de Estimulação Magnética Transcraniana, permitindo-lhes visualizar sobre a cabeça dos pacientes o modelo do cérebro dos mesmos, com a área a estimular assinalada, de modo a que os médicos consigam mais facilmente perceber onde devem aplicar a estimulação. Apesar deste uso específico, toda a aplicação pode ser útil a médicos – na aplicação da EMT e não só – e também a estudantes de Medicina – como apoio ao estudo do cérebro – e a outros interessados na visualização e interação em RA e RV com modelos virtuais do cérebro.

Neste capítulo discutem-se as razões para a baixa adoção das aplicações de RA na Medicina, as vantagens da solução proposta e, tendo em conta os resultados dos testes de utilizador, os problemas que esta ainda apresenta, as resoluções que se tentaram, os motivos para não terem resultado e outros aspetos a melhorar. A partir dessa discussão e também das sugestões recolhidas nos testes de utilizador, apresenta-se então uma listagem de potencial trabalho futuro e, em seguida, fazem-se algumas observações finais sobre este projeto.

7.1 Discussão

Foi feita uma análise do estado-da-arte das aplicações de RA na Medicina e verificou-se que já existem muitas provas de conceito na área, algumas delas muito avançadas. Contudo, a adoção real ainda é muito baixa e tem acontecido lentamente, devido a diversos fatores:

- custos elevados das soluções;
- limitações nos equipamentos disponíveis;
- precisão ainda insuficiente do *tracking*;
- visualizações muito confusas, com muita informação ao mesmo tempo, ou pouco realistas;
- necessidade de *displays* e/ou equipamentos adicionais pouco confortáveis ao utilizador;
- dificuldades em enquadrar o sistema de RA no local necessário e no fluxo de trabalho tradicional;
- desconhecimento e desconfiança dos utilizadores em relação às vantagens reais da adoção destes sistemas.

A solução desenvolvida distingue-se em alguns destes aspetos, constituindo uma solução de baixo custo e bastante acessível, por o único equipamento necessário ser um *smartphone*. Este facto também faz com que a curva de aprendizagem e dificuldade de adaptação sejam reduzidas, pois é um dispositivo móvel com o qual a maior parte dos médicos está já habituada a trabalhar. É também um dispositivo pequeno, confortável de segurar, apesar de que seria uma vantagem ter uma solução que não ocupasse uma das mãos do utilizador.

A vantagem referida acima poderá ser conseguida com a utilização de óculos de RA. Durante o projeto estudou-se um modo de uso da aplicação com o Durovis Dive [33], um suporte para *smartphone* que permite usar este equipamento como se se tratasse de uns óculos de RA/RV. Esta é uma solução de baixo custo, uma vez que se trata apenas de um suporte em que o *smartphone* é colocado à frente dos olhos do utilizador, sendo o dispositivo móvel a encarregar-se de todo o poder de computação necessário. Contudo, o uso deste equipamento implicava ter a aplicação a funcionar em modo estereoscópico,

com o ecrã dividido, o que não proporcionava uma experiência agradável porque a imagem da câmara aparecia igual nos dois ecrãs, por o telemóvel só ter uma câmara. No final do desenvolvimento deste projeto obteve-se acesso a um novo conjunto de óculos, mais caro, os ORA da Optinvent [50], em que o computador vem integrado nos óculos, funcionando também com o sistema operativo Android. Assim, será possível instalar a aplicação *Brain AR/VR* nos óculos, no entanto serão necessários fazer ajustes no código para que a experiência seja adequada ao novo *display* – que consiste num pequeno ecrã que fica em frente ao olho direito – e à diferente forma de interação com o ecrã – um *touchpad* existente na lateral dos óculos. Estes novos óculos não ficaram disponíveis a tempo de se poder instalar, ajustar e testar a aplicação nos mesmos ainda dentro do período de desenvolvimento do projeto, pelo que essas tarefas são deixadas como trabalho futuro.

A aplicação foi testada por utilizadores da área da Medicina, Engenharia Biomédica e Física, alguns dos quais médicos e alguns dos quais experientes em EMT. A apreciação dos utilizadores foi bastante positiva e para além dos resultados quantitativos permitiu obter sugestões de mudanças, melhorias e funcionalidades a incorporar. No geral, os resultados vão de encontro à avaliação feita pelos participantes no desenvolvimento do projeto, inclusive nos aspetos mais importantes que ainda precisam de ser melhorados no futuro para o bom desempenho da aplicação, em particular a estabilidade dos modelos virtuais no modo de RA e o seu realismo.

O problema da estabilidade dos modelos virtuais prende-se com o facto de a posição/orientação do modelo do cérebro oscilar muito sobre o mesmo ponto, sob certas condições. Isto deve-se a erros de aproximação na deteção das posições/orientações das marcas, especialmente se luminosidade forte estiver constantemente a perturbar a deteção das marcas pelo *software*. A instabilidade dos modelos agrava-se com a distância da câmara às marcas, sendo que a distâncias curtas o modelo apresenta-se bastante constante.

Além disto, e pelos mesmos problemas na deteção das marcas, o processo de calibração não é totalmente exato, sendo que os referenciais das várias marcas, devendo coincidir, não ficam totalmente coincidentes. Em condições de fraca deteção das marcas ou, por exemplo, devido a movimentos bruscos, a marca usada em cada instante para alinhar o modelo virtual está constantemente a mudar – em casos em que mais que uma marca é visível – ou então o modelo desaparece nos instantes em que a deteção falha.

Como a coincidência dos referenciais das diferentes marcas não é exata, as diferenças no alinhamento do modelo notam-se sempre que a detecção de uma marca falha e a aplicação passa a usar outra para alinhar o modelo, sendo este outro motivo para a instabilidade.

Normalmente a diferença entre os referenciais é aceitável, mas por vezes algumas marcas ficam com uma calibração totalmente errada, levando a que, nos instantes em que essas marcas são usadas para o alinhamento, os modelos apareçam visivelmente fora da posição/orientação em que deviam estar. Nestes casos, o procedimento sugerido de momento é detetar qual a marca que ficou mal calibrada e removê-la da touca, o que não proporciona grande inconveniente quando apenas se retira uma marca. Contudo, como trabalho futuro é necessário resolver o problema ou minimizar os seus efeitos.

Uma ideia já tentada foi possibilitar a recalibração das marcas, mas os erros mantinham-se. Também se tentou, devido a todas estas limitações do uso de marcas, usar uma nova abordagem de RA, através dos sensores de movimento do *smartphone*, mas os resultados não foram satisfatórios e os erros no *tracking* eram inaceitáveis. Talvez com mais tempo fosse possível chegar a uma implementação que conseguisse conjugar sensores e marcas para uma melhor experiência de RA, mas já não foi possível continuar este estudo no período de desenvolvimento do projeto, pelo que é deixado como potencial trabalho futuro.

No que diz respeito ao realismo dos modelos do cérebro, é possível conseguir melhores resultados através da introdução de fontes de luz virtuais junto ao modelo, que farão com que sombras apareçam sobre o modelo e destaquem os seus contornos e relevos. Inicialmente tentou-se incluir essas sombras nas texturas do modelo do cérebro, através de um processo no Blender chamado “Baking” de texturas, o que permitiria que a informação sobre luz e sombra já estivesse incorporada nos modelos previamente ao seu uso pela aplicação. Não seria, pois, necessário que a aplicação introduzisse as fontes de luz virtuais e calculasse o efeito em termos de sombra no modelo, algo que é possível de efetuar com o Metaio SDK, mas que obriga a um maior poder de processamento do *smartphone*. Contudo, existiram problemas em fazer o “baking” das texturas no Blender e optou-se por deixar os modelos com uma representação mais simples, usando apenas cores lisas enquanto texturas, e esperar pelo *feedback* dos utilizadores na matéria. Vários

destes utilizadores demonstraram considerar necessária a inclusão de mais realismo nos modelos, pelo que essa inclusão é deixada como trabalho futuro.

O funcionamento da aplicação está dependente da preparação e introdução de alguns ficheiros no *smartphone*, em particular de um ficheiro de configuração do *tracking*, um ficheiro de definição das geometrias (os vários submodelos em que o modelo do cérebro está dividido, de acordo com a parcelização em áreas) e os ficheiros que contém os vários submodelos correspondentes às diferentes áreas do cérebro. Estes últimos, em particular, devem ser idealmente obtidos a partir de imagens de ressonância magnética do paciente, através de um processo que pode constituir uma dificuldade ou pelo menos um gasto de algum tempo para o médico, algo que seria útil minimizar.

No que toca à forma como a aplicação reconhece esses ficheiros na memória do telemóvel, baseia-se no nome e localização dos ficheiros, o que poderá ser substituído no futuro por um reconhecimento mais amigável ao utilizador, através de uma interface inicial na aplicação que permita seleccionar quais os ficheiros a ler. Também como trabalho futuro se poderá estender essa interface e as funcionalidades da aplicação para que seja possível manter uma lista de pacientes, com as localizações dos ficheiros que lhes estão associados, o que permitiria um uso muito mais simples da aplicação com pacientes que já estivessem guardados na lista.

7.2 Trabalho Futuro

Apresenta-se em seguida uma lista de tarefas (organizadas em diversas secções) que podem ser realizadas no futuro para melhorar/estender a aplicação:

Modo de RA:

- Estudar novos formatos (mais gráficos) para os indicadores de calibração, ou alterações de cor/tamanho aos existentes;
- Acrescentar um sinal sonoro quando uma marca é calibrada;
- Alterar a identificação das marcas, por exemplo, adotando as posições e nomenclatura utilizada em EEGs;

- Melhorar a estabilidade do alinhamento do modelo do cérebro, colocando, por exemplo, alguma inércia no modelo, de forma não responder a todos os estímulos mínimos;
- Solucionar o problema das calibrações incorretas ou estudar melhores alternativas para efetuar o *tracking* da cabeça do paciente;
- Estudar a possibilidade de remoção da indicação colorida da distância do aparelho de estimulação ao cérebro;
- Melhorar os dois glifos (e potencialmente incluir mais) para que permitam visualizar fielmente o sentido e a zona de intensidade máxima do campo magnético na bobina do aparelho de estimulação, para os diferentes formatos de bobinas/aparelhos utilizados;
- Adicionar transparência aos glifos do aparelho de estimulação, para que se visualize o modelo virtual do cérebro por detrás;

Modo de RV:

- Alterar o gesto usado para rodar o modelo do cérebro;
- Aumentar tamanho do texto dos botões de escolha do eixo de rotação;
- Possibilitar a translação do modelo ou pelo menos subir a posição inicial do mesmo, para que não fique oculto pelos botões da interface;
- Alterar os nomes das vistas predefinidas nos botões da interface para nomes utilizados na neurologia;
- Possibilitar a personalização do tamanho/posição/orientação inicial do modelo;

Visualização/Interface:

- Aumentar o realismo dos modelos do cérebro, através, por exemplo, da inclusão de iluminação virtual;
- Inserir maior variedade de cores/tons nas áreas do cérebro, para que se distingam melhor, e estudar que cores utilizar nessas áreas e na área (ou áreas) que se quer destacar;
- Permitir a alteração, na aplicação, da área (ou áreas) a destacar;

- Adicionar realce aos botões selecionados na interface, especialmente na barra de botões de escolha do eixo de rotação no modo de RV;
- Incluir no menu de opções uma caixa de verificação para marcar/desmarcar automaticamente todas as áreas;
- Colocar alguma transparência nas áreas desmarcadas no menu de opções, em vez de estarem totalmente invisíveis;
- Adicionar um botão e ecrã de ajuda;
- Possibilitar a escolha do idioma da aplicação (pelo menos entre português e inglês);

Preparação e Introdução de Informação:

- Facilitar ou autonomizar o processo de obtenção dos modelos e de preparação dos restantes ficheiros;
- Criar uma interface inicial na aplicação para seleção dos ficheiros a ler;
- Possibilitar a manutenção, pela aplicação, de uma lista de pacientes, com o registo das localizações na memória do telemóvel dos ficheiros a carregar associados a cada um;

Outras extensões:

- Estudar e implementar soluções para novas visualizações, como por exemplo de posições de elétrodos intracranianos ou de mapas funcionais;
- Instalar, adaptar e testar a aplicação em óculos de RA, como os ORA da Optinvent;

7.3 Conclusão

Deste projeto resultou uma aplicação completamente funcional, testada por utilizadores, que cumpre os requisitos propostos. Foram também efetuadas várias demonstrações públicas da aplicação. Resultou ainda um artigo científico aceite numa conferência internacional, que descreve a aplicação criada. Assim, pode-se afirmar que os objetivos do projeto foram cumpridos.

A aplicação, como já referido, tem ainda algumas arestas que precisam de ser limadas, especialmente no que diz respeito ao *tracking* da cabeça do paciente, no modo de RA, devido às limitações na deteção precisa das marcas e calibração das mesmas. Contudo, o modo de RA funciona e a aplicação é uma prova de conceito de que é possível utilizar equipamento acessível, como um *smartphone*, para ter uma experiência de visualização do cérebro em RA e RV, com funcionalidades que podem ser bastante úteis para utilizadores na área da Medicina.

É importante referir que esta aplicação poderia ser aplicada a outras áreas, podendo com relativa facilidade ser adaptada para a visualização em RA e RV de outros órgãos do corpo humano, por exemplo. Além disso, enquanto ferramenta para visualização do cérebro num *smartphone*, a aplicação *Brain AR/VR* tem grandes potencialidades de extensão, podendo no futuro ser aumentada para permitir outras visualizações de dados cerebrais, com aplicações médicas para além da EMT, tornando-se num instrumento versátil de visualização do cérebro.

Anexo A: Manual de Utilizador

Aplicação *Brain AR/VR*

A aplicação *Brain AR/VR* proporciona experiências de visualização de modelos virtuais do cérebro em Realidade Aumentada (RA) e Realidade Virtual (RV). Os modelos virtuais têm de ser obtidos a partir de imagens de ressonância magnética, de acordo com um processo explicado neste manual. O modo de RA da aplicação foi pensado para guiar médicos na aplicação da técnica de Estimulação Magnética Transcraniana (EMT). No entanto, toda a aplicação pode ser útil para estudantes de Medicina ou qualquer pessoa que tenha interesse em visualizar o cérebro e a sua organização em diferentes áreas.

Interface geral

A aplicação fornece dois modos de uso: RA e RV. Pode escolher entre os dois modos carregando em dois botões (“AR” e “VR”, respetivamente) que estão presentes na interface de utilizador (Figura A.1). Existe ainda um terceiro botão (“Options”) para entrar num menu de opções, que pode ser acedido a partir de ambos os modos. Os três botões apresentam-se no canto inferior direito da aplicação e, para não ocupar espaço de visualização, desaparecem após alguns segundos sem toques no ecrã. Para que os botões voltem a aparecer basta tocar em qualquer ponto do ecrã. Existe ainda um botão (com uma cruz) para sair da aplicação, no canto superior direito, que se mantém sempre visível na interface.

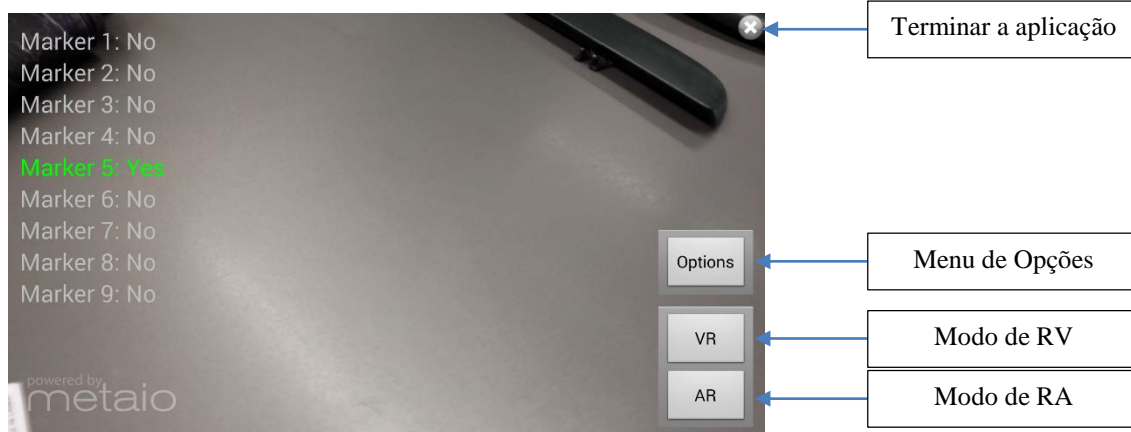


Figura A.1 – Interface de utilizador da aplicação

Modo de RA

O modo de RA permite visualizar modelos virtuais 3D do cérebro alinhados sobre a cabeça de um indivíduo (paciente). Idealmente, e para o contexto da EMT, os modelos representarão o cérebro do próprio indivíduo sobre o qual se aplica a técnica, tendo sido previamente obtidos a partir de imagens de ressonância magnética do indivíduo, trabalho prévio que tem de ser realizado pelo utilizador médico (ver secção “Obtenção dos modelos”).

A experiência de RA é proporcionada graças ao uso de marcas colocadas numa touca na cabeça do paciente, marcas que a câmara do *smartphone* consegue reconhecer e usar para alinhar os modelos virtuais sobre a cabeça do paciente. A aplicação reconhece uma configuração com 9 marcas de 15 mm, numeradas de 1 a 9 (Figura A.2). A touca contém velcros a marcar as posições onde as marcas devem ser colocadas. A marca 5 deve ser colocada no topo da cabeça, enquanto a posição das outras marcas não precisa de seguir uma organização específica.

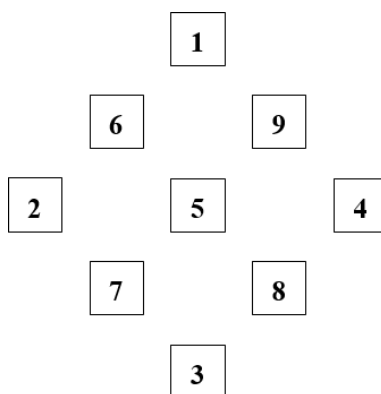


Figura A.2 – Esquema de configuração das marcas

Para poder usufruir do modo de RA, deve passar primeiro por um processo de calibração das marcas, necessário para que o alinhamento dos modelos virtuais fique correto independentemente de qual a marca que a câmara está a detetar. A marca 5 deve ser calibrada previamente no ficheiro de configuração (ver secção “Ficheiro de configuração”). As restantes marcas devem ser calibradas ao início do uso do modo de RA, existindo indicadores visuais para que saiba quais as marcas que ainda precisa de calibrar (ver secção “Calibração das marcas”).

Calibração das marcas

Utilize os indicadores no canto superior esquerdo do ecrã para saber em cada momento quais as marcas que já estão calibradas e quais as que ainda falta calibrar. Os indicadores apresentam texto da forma “Marker N: R”, em que “N” é o número da marca e “R” é igual a “Yes” se a marca já estiver calibrada e “No” se a marca ainda não estiver calibrada. O texto de marcas calibradas apresenta-se a verde, permitindo que se distingam mais facilmente as marcas calibradas das não-calibradas. As marcas possuem um número escrito para que possa fazer a associação aos números mostrados nos indicadores.

Para calibrar uma marca:

- Aponte a câmara do *smartphone* de forma a captar na imagem ao mesmo tempo uma marca calibrada e uma marca não-calibrada;
- Certifique-se de que as “Condições de deteção das marcas” estão a ser verificadas (ver secção com esse nome);
- Mantenha o *smartphone* apontado para as duas marcas até que o indicador relativo à marca não-calibrada esteja verde.

Quando todas as marcas estiverem calibradas aparece por baixo dos indicadores a mensagem “ALL MARKERS CALIBRATED!!!” e os modelos virtuais ficam visíveis (ver Figura A.3). Segundos depois, todos os indicadores e a mensagem desaparecem, para facilitarem a visualização.



Figura A.3 – Ecrã do modo de RA durante (esquerda) e no final (direita) do processo de calibração das marcas

ATENÇÃO: Em caso de má deteção das marcas durante o processo de calibração, algumas das marcas podem ficar mal calibradas, o que fará com que o alinhamento dos modelos do cérebro apareça errado durante a execução da aplicação quando a câmara deteta alguma das marcas mal calibradas. Solucione o problema removendo as marcas que provocam o desalinhamento ou então reiniciando a aplicação e executando a calibração com calma e em boas condições de deteção das marcas.

Condições de deteção das marcas

A deteção das marcas é essencial para o funcionamento do modo de RA, sendo uma componente bastante sensível da aplicação, pois a deteção de cada marca durante o processo de reconhecimento de imagem feito pela aplicação está sujeito a uma captação correta da forma, cor e conteúdo de cada marca. Existem portanto alguns fatores a ter em conta para uma correta deteção das marcas:

- Visibilidade total: Toda a marca (delimitada pelo rebordo preto) tem de estar visível na imagem da câmara;
- Condições de luminosidade: Luz muito fraca ou luz muito forte a incidir numa marca impede que a câmara a reconheça. Observe a imagem da câmara para perceber se as marcas estão a ser bem captadas e experimente alterar a incidência ou intensidade da luz (ou colocar a cabeça do paciente noutra ângulo) até que as marcas consigam ser bem captadas;
- Distância do *smartphone* às marcas: Usualmente o reconhecimento das marcas funciona a distâncias inferiores a 70 cm, podendo esse limite aumentar ou diminuir em função dos restantes fatores de influência na deteção. Uma configuração com marcas maiores permite também aumentar o limite para a distância, mas com desvantagens (ver secção “Ficheiro de configuração”);
- Ângulo de captação: Marcas captadas na diagonal, em ângulos muito inclinados, aparecem de forma distorcida na imagem da câmara, o que impede o seu reconhecimento. Se necessário, altere o posicionamento do *smartphone* de modo a que as marcas sejam detetadas.

Para o funcionamento da experiência de RA basta que em cada instante uma marca esteja a ser detetada. Durante a fase inicial de calibração, porém, duas marcas precisam de ser detetadas em cada instante (ver secção “Calibração das marcas”).

Glifo do aparelho de estimulação

A aplicação permite, no modo de RA, a representação do aparelho de estimulação utilizado na EMT, desde que seja colocada uma marca específica também sobre esse aparelho. Essa representação é um glifo, uma simplificação do aspeto real do aparelho de estimulação e serve apenas para guiar o utilizador e

melhorar a percepção de profundidade. O glifo tem um código de cores associado, sendo vermelho quando o aparelho de estimulação está a grande distância do cérebro, amarelo quando a distância é intermédia e verde quando próximo. Existem duas representações alternativas para o glifo, entre as quais o utilizador pode escolher no menu de opções (ver Figura A.4).

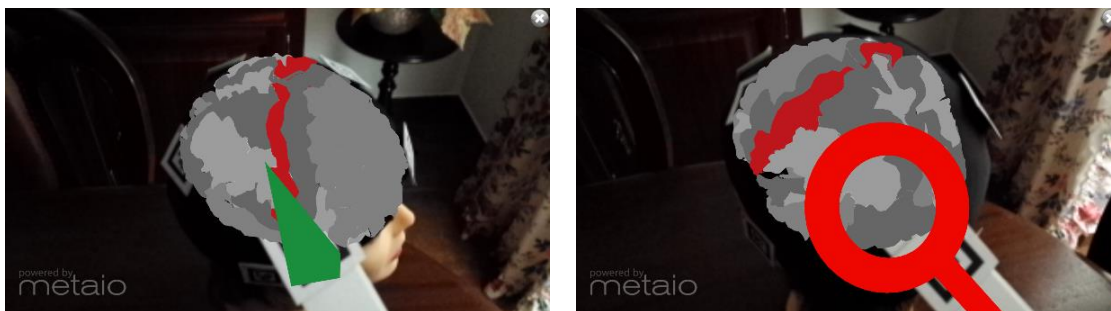


Figura A.4 – Glifos do aparelho de estimulação: Arrow (esquerda) e Coil (direita)

Para que o glifo do aparelho de estimulação seja visível, a marca 10 deve ser colada sobre o aparelho, na intersecção entre o cabo e a zona principal. Certifique-se que a marca se mantém virada para a câmara durante o procedimento, caso contrário a visualização pode falhar. Tal como com as marcas da touca, tenha em atenção os cuidados a ter para a deteção das marcas, descritos na secção “Condições de deteção das marcas”. Além disso, para não interferir com a visualização do cérebro, garanta que o aparelho de estimulação não tapa todas as marcas da touca que a imagem da câmara está a detetar, existindo sempre uma completamente visível em cada instante.

Modo de RV

O modo de RV permite visualizar os modelos virtuais do cérebro parados no ecrã, sem ligação à imagem da câmara, que fica oculta, para que consiga controlar a orientação dos modelos e observá-los com maior ou menor escala. A interação é feita por toque, estando dois gestos definidos:

- Colocar dois dedos no ecrã e afastá-los ou aproximá-los para, respetivamente, aumentar ou diminuir a escala dos modelos;
- Colocar dois dedos no ecrã e movê-los num gesto circular para rodar os modelos num sentido ou noutro, de acordo com um determinado eixo de rotação.

O eixo de rotação pode ser alterado através de dois botões (“Horiz.” e “Vertic.”) colocados na zona inferior central da interface. À esquerda existem seis outros botões que permitem selecionar vistas predefinidas, isto é, rodar o modelo automaticamente para se adequar a essas vistas: frente (“Front”), trás (“Back”), esquerda (“Left”), direita (“Right”), cima (“Top”) e baixo (“Bottom”). Estes dois grupos de botões são exclusivos ao modo de RV e, tal como os botões para escolha de modo e entrar no menu de opções, desaparecem após alguns segundos sem toques no ecrã, reaparecendo ao tocar. A interface do modo de RV é apresentada na Figura A.5.

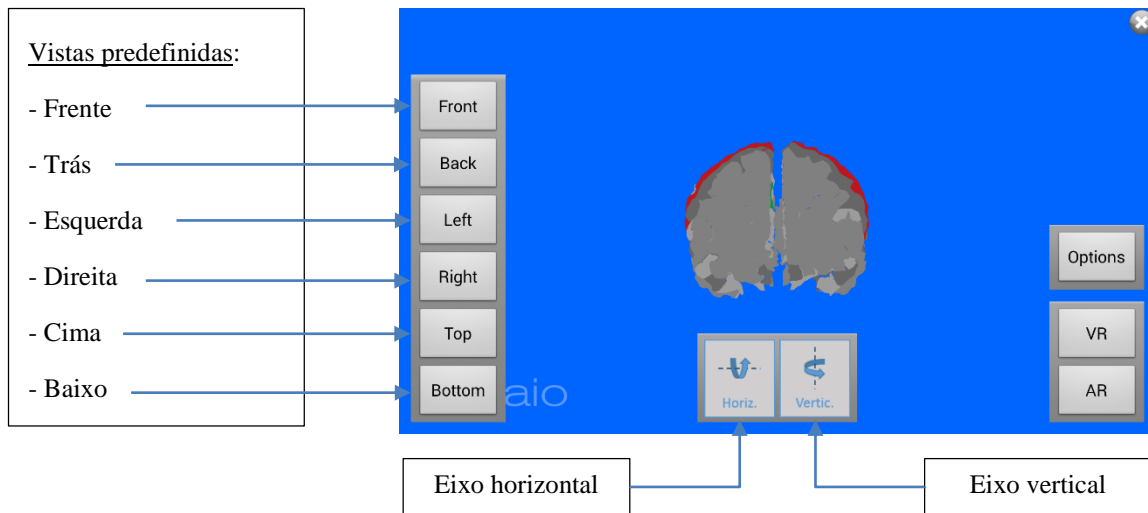


Figura A.5 – Interface de utilizador do modo de RV

Menu de Opções e Visualização Seletiva

A aplicação oferece uma funcionalidade de Visualização Seletiva – que se aplica tanto ao modo de RA como ao de RV – que lhe permite escolher que áreas do cérebro pretende visualizar, ao invés de visualizar o cérebro completo. A escolha das áreas é feita no menu de opções, que contém uma caixa de verificação para cada área, que pode marcar ou desmarcar para visualizar ou ocultar, respetivamente, essa área. Caixas de verificação relativas aos dois hemisférios e quatro lobos permitem-lhe ainda que marque ou desmarque mais rapidamente todas as áreas de um dos hemisférios ou lobos.

No menu de opções pode alterar-se o glifo usado para representar o aparelho de estimulação, entre uma de duas representações (ver secção “Glifo do aparelho de estimulação”). A interface do menu de opções é apresentada na Figura A.6.

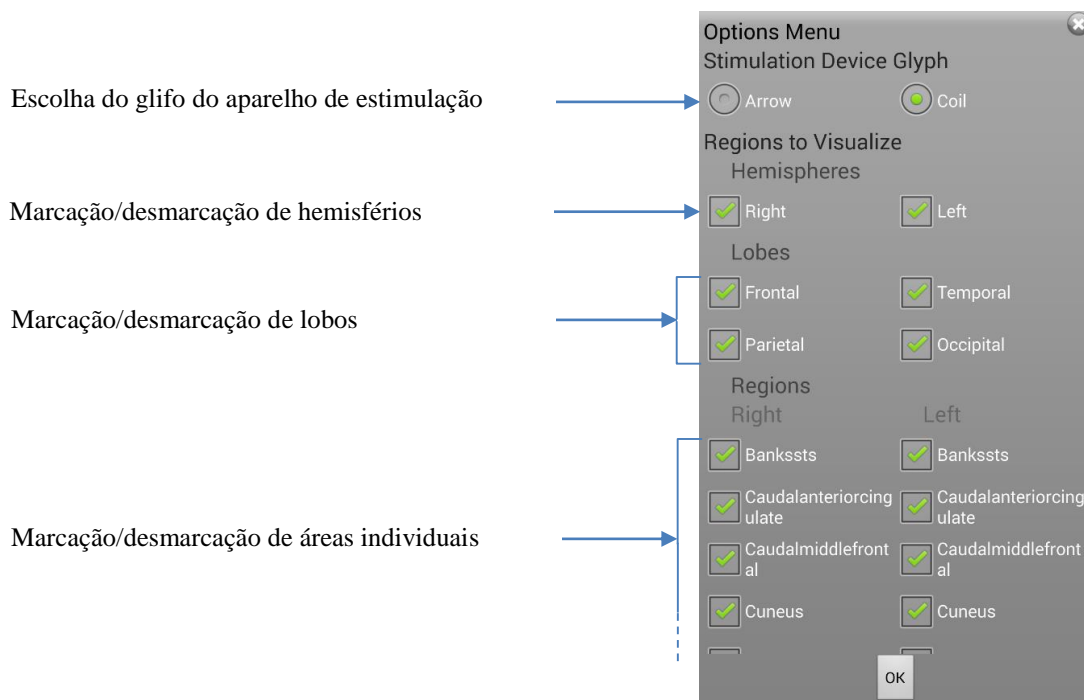


Figura A.6 – Interface de utilizador do menu de opções

ATENÇÃO: O menu tem mais opções do que as que conseguem ser apresentadas inicialmente no ecrã, podendo ser necessário deslizar para percorrer todo o menu. Usar o *smartphone* na orientação vertical permite que seja apresentada mais informação no ecrã.

Preparação Prévia

O funcionamento da aplicação necessita de alguma informação externa que deve ser preparada pelo utilizador. Em particular, é preciso obter os modelos virtuais do cérebro no formato “.obj” e preencher um ficheiro de definição (formato “.txt”) com algumas informações sobre os mesmos. Para além disso, é necessário adicionar um ficheiro de configuração (formato “.xml”) que é usado no modo de RA e que pode ter de ser modificado quando se muda de paciente (o indivíduo sobre o qual o modelo do cérebro deve ser alinhado). A informação sobre cada um desses elementos e como devem ser guardados no *smartphone* é dada nas secções seguintes.

Ficheiro de configuração

O ficheiro de configuração, no formato “.xml”, guarda informação sobre as marcas a detetar pela aplicação e permite calibrar a marca 5, a do topo da cabeça, a única que é calibrada previamente.

A marca contém um referencial cartesiano ortogonal, ou seja, um sistema de três eixos perpendiculares entre si (‘x’, ‘y’ e ‘z’). A direção e sentido dos eixos ‘x’ e ‘y’ é indicada na própria marca, enquanto o eixo ‘z’ é perpendicular ao plano da marca e tem o sentido a apontar “para cima”, para fora da marca. A intersecção dos três eixos, no centro da marca, chama-se a origem do referencial.

Os modelos virtuais do cérebro são representados também centrados num referencial, cuja origem deve estar num ponto interior central da cabeça do paciente. A calibração da marca consiste em indicar qual a relação existente entre o referencial da marca e o referencial dos modelos do cérebro. Essa relação é dada em termos de uma translação (o movimento que é feito da origem do referencial da marca para a origem do referencial do cérebro) e possivelmente de uma rotação (quanto é que o sistema de eixos da marca deve rodar para ficar com a mesma orientação que o sistema de eixos do cérebro) (ver Figura A.7).

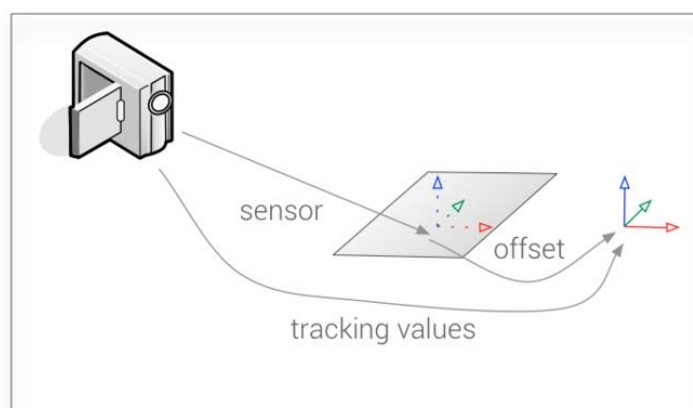


Figura A.7 – Esquema de aplicação de uma translação e/ou rotação ao referencial de uma marca [5]

A translação é medida em milímetros reais e é dada por três valores, indicando o deslocamento que a origem do referencial da marca deve sofrer segundo cada um dos eixos até coincidir com a origem do referencial pretendido para o cérebro. É possível evitar dar valores à rotação se a marca for colocada de modo a que o seu referencial já tenha a mesma orientação que o referencial do cérebro.

Aconselha-se então que a marca 5 seja colocada no topo da cabeça com o eixo ‘y’ a apontar para a parte de trás da cabeça do paciente, o eixo ‘z’ a apontar para cima e, conseqüentemente, o eixo ‘x’ a apontar para a esquerda do paciente. Durante o processo de obtenção dos modelos virtuais, eles devem ser guardados seguindo essas mesmas orientações para os eixos (ver secção “Obtenção dos modelos”).

No ficheiro de configuração, a calibração é feita dentro do elemento <TrackingData> ---> <Connections> ---> <COS> ---> <SensorSource> ---> <COSOffset>. Existem vários elementos <SensorSource>, sendo o correto aquele cujo nome (dado pelo elemento <SensorCosId>) é “Marker5”. Dentro do elemento <COSOffset> existem dois elementos: <TranslationOffset> e <RotationOffset>, que permitem aplicar o deslocamento (em termos de translação e rotação, respetivamente) ao referencial da marca 5 (ver Figura A.8).

O elemento <TranslationOffset> contém três elementos <X>, <Y> e <Z>, onde se devem inserir as medidas (em milímetros) de deslocamento segundo cada um dos eixos para que o referencial da marca fique posicionado num ponto central interior da cabeça do paciente, para que o modelo do cérebro apareça na posição correta durante o uso da aplicação.

```
<SensorSource>
  <SensorID>Markertracking1</SensorID>
  <SensorCosID>Marker5</SensorCosID>
  <HandEyeCalibration>
  <COSOffset>
    <TranslationOffset>
      <X>0</X>
      <Y>0</Y>
      <Z>-70</Z>
    </TranslationOffset>
    <RotationOffset>
      <X>0</X>
      <Y>0</Y>
      <Z>0</Z>
      <W>1</W>
    </RotationOffset>
  </COSOffset>
</SensorSource>
```

Figura A.8 – Exemplo do conteúdo de um elemento <COSOffset> num ficheiro de configuração

Obtenção dos modelos

O modelo virtual do cérebro do paciente deve ser obtido através de técnicas de imagiologia médica, como ressonância magnética. Durante o processo é necessário fazer a parcelização do cérebro em diferentes áreas, de acordo com um atlas cerebral desejado. Em seguida descreve-se um processo de obtenção do modelo virtual a partir de imagens obtidas por ressonância magnética:

1. Obtenha as imagens em formato DICOM, ponderadas em T1;
2. Converta as imagens para o formato NIfTI, numa matriz de intensidade de sinal;
3. Com o *software* Freesurfer faça a segmentação do cérebro em substância cinzenta, branca e líquido cefalorraquidiano, com uma divisão entre as regiões corticais e as subcorticais;
4. Realize a parcelização do cérebro, com base num atlas anatómico desejado, separando as áreas da parcelização em ficheiros individuais;
5. Converta os ficheiros para o formato OBJ, que pode ser lido por *software* de modelação como o Blender. Para tal, utilize as ferramentas disponibilizadas pelo projeto Brainder (<http://brainder.org/download/brain-for-blender/>);
6. Se necessário para melhorar o desempenho da aplicação, utilize o Blender para diminuir o número de polígonos dos modelos de cada área. Para tal, pode basear-se no script Blender Python disponibilizado no Anexo G.

Ficheiro de definição

O ficheiro de definição é um ficheiro de texto lido pela aplicação para saber quais os modelos que deve carregar (um modelo para cada área), com que cor devem ser representados e como parte de que hemisfério e lobo devem ser considerados. Assim, crie o ficheiro num computador e preencha-o da seguinte forma:

- Primeira linha:

“(n° áreas total) (n° áreas hemis. esquerdo) (n° áreas hemis. direito) (n° áreas lobo frontal) (n° áreas lobo temporal) (n° áreas lobo parietal) (n° áreas lobo occipital)”

- Cada uma das linhas seguintes corresponde à informação sobre uma das áreas, tendo cada linha o seguinte formato:

“(caminho do ficheiro com o modelo da área) (hemisfério da área) (lobo da área) (cor para representar a área)”

- O caminho dos ficheiros na memória do *smartphone* deve ser dado relativamente à pasta “BrainARVR”, uma pasta que deve estar dentro da pasta raiz do armazenamento do dispositivo móvel (ver secção “Introdução da informação na aplicação”)
- O hemisfério deve ser indicado como:
 - “L”, se for o esquerdo,
 - “R”, se for o direito.
- O lobo deve ser indicado como:
 - “F”, se for o frontal,
 - “T”, se for o temporal,
 - “P”, se for o parietal,
 - “O”, se for o occipital
- A cor de cada área deve ser indicada entre uma das seguintes:
 - “Grey25”, um tom mais claro de cinzento,
 - “Grey37,5”, um tom intermédio de cinzento,
 - “Grey50”, um tom mais escuro de cinzento,
 - “Red”, um tom mais claro de vermelho,
 - “Red2”, um tom intermédio de vermelho,
 - “Red1”, um tom mais escuro de vermelho,
 - “Green”, um tom de verde,

- “Blue”, um tom de azul,
- “Cyan”, um tom de ciano,
- “Magenta”, um tom de magenta,
- “Yellow”, um tom de amarelo

Um exemplo de um ficheiro de definição pode ser encontrado no Anexo F, encontrando-se na Figura A.9 um excerto desse exemplo.

```
70 35 35 26 20 14 10
white_DK_obj_dec/lh.white.DK.bankssts.obj L T Grey25
white_DK_obj_dec/lh.white.DK.caudalanteriorcingulate.obj L F Grey25
white_DK_obj_dec/lh.white.DK.caudalmiddlefrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.cuneus.obj L O Grey37,5
white_DK_obj_dec/lh.white.DK.entorhinal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.frontalpole.obj L F Grey50
white_DK_obj_dec/lh.white.DK.fusiform.obj L T Grey25
white_DK_obj_dec/lh.white.DK.inferiorparietal.obj L P Grey25
white_DK_obj_dec/lh.white.DK.inferiortemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.insula.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.isthmuscingulate.obj L P Grey50
white_DK_obj_dec/lh.white.DK.lateraloccipital.obj L O Grey50
white_DK_obj_dec/lh.white.DK.lateralorbitofrontal.obj L F Grey25
white_DK_obj_dec/lh.white.DK.lingual.obj L O Grey25
white_DK_obj_dec/lh.white.DK.medialorbitofrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.middletemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.paracentral.obj L F Grey50
white_DK_obj_dec/lh.white.DK.parahippocampal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.parsopercularis.obj L F Grey25
white_DK_obj_dec/lh.white.DK.parsorbitalis.obj L F Grey25
white_DK_obj_dec/lh.white.DK.parstriangularis.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.pericalcarine.obj L O Grey37,5
white_DK_obj_dec/lh.white.DK.postcentral.obj L P Red
white_DK_obj_dec/lh.white.DK.posteriorcingulate.obj L P Grey50
white_DK_obj_dec/lh.white.DK.precentral.obj L F Grey50
white_DK_obj_dec/lh.white.DK.precuneus.obj L P Grey25
```

Figura A.9 – Excerto de um exemplo de um ficheiro de definição

Introdução da informação na aplicação

A aplicação precisa de ler o ficheiro de configuração (“.xml”), o ficheiro de definição (“.txt”) e os ficheiros que contém os modelos das várias áreas (“.obj”). Para isso, esses ficheiros devem ser guardados numa pasta de nome “BrainARVR” (criar, se não existir). Esta pasta deve estar dentro da pasta raiz do armazenamento do dispositivo (a mesma pasta onde por predefinição se encontram as pastas “Alarms”, “DCIM”, “Downloads”, “Movies”, “Music”, “Notifications”, “Pictures”, “Podcasts” e “Ringtones”, criadas pelo sistema operativo).

- O nome do ficheiro de configuração deve ser “trackingConfig.xml”.
- O nome do ficheiro de definição deve ser “geometryDef.txt”.
- Os modelos podem ter qualquer nome de ficheiro, pois as suas localizações devem ser incluídas no ficheiro de definição.
 - Exemplo: Um modelo com nome “model_temporal_pole.obj” que esteja guardado dentro de uma pasta “ModelsPatientA” que esteja dentro da pasta “BrainARVR” deve ser indicado no ficheiro de definição com o caminho “ModelsPatientA/model_temporal_pole.obj”

Anexo B: Arquitetura da aplicação e fluxo de informação associado

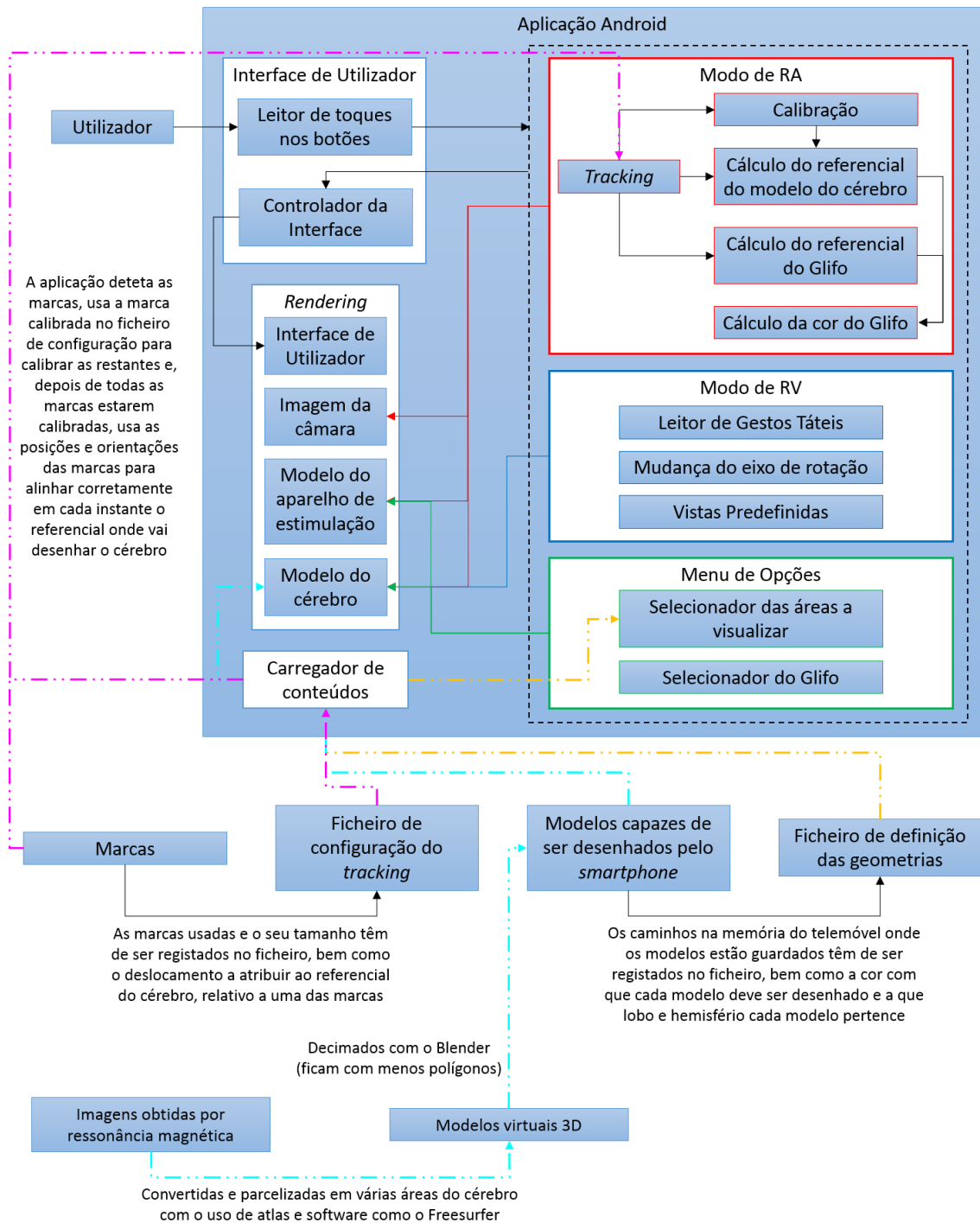


Figura B.1 – Arquitetura da aplicação e fluxo de informação associado

Anexo C: Cálculos associados à calibração das marcas

A calibração de uma marca consiste no cálculo de uma transformação (*offset*) em termos de translação e rotação a atribuir ao referencial dessa marca de modo a que este coincida com o referencial de uma marca já calibrada. Assim, considere-se como i o ID da marca que se deseja calibrar e j o ID da marca já calibrada que está a ser usada para o processo. A posição e orientação atuais dos referenciais de cada marca são obtidos através do *tracking* das marcas e guardados na forma de objetos “TrackingValues”, também chamados poses, que englobam um objeto “Vector3D” de translação e um objeto “Rotation” de rotação em relação ao referencial da câmara. Seja tv_i a pose da marca i em relação à câmara e tv_j a pose da marca j em relação à câmara. O objetivo da calibração é calcular a pose tv_k (translação e rotação) que deve ser usada para transformar o referencial da marca i no referencial da marca j (ver Figura C.1).

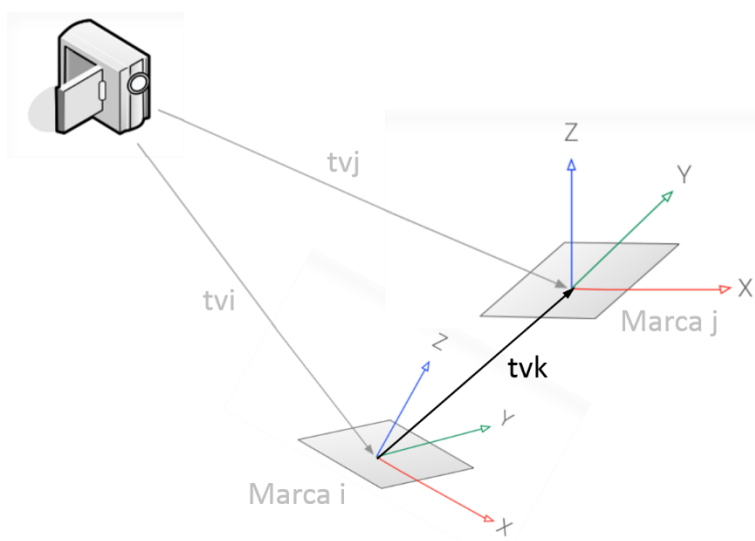


Figura C.1 – Esquema ilustrativo do funcionamento do processo de calibração de uma marca

Numa primeira aproximação ao problema, pensou-se no cálculo simples:

$$tv_k.trans = (-tv_i.trans) + tv_j.trans$$

$$tv_k.rot = (-tv_i.rot) * tv_j.rot,$$

em que *trans* e *rot* correspondem às translações e rotações, respetivamente, das poses sobre as quais são chamados, $-$ denota a operação de simetria/inversão e $*$ denota a operação de multiplicação (ou composição) de rotações. A operação de inversão de uma translação/rotação corresponde a obter a translação/rotação que vai no sentido contrário, até ao antigo referencial de partida, partindo do antigo referencial de chegada e estando escrita nas coordenadas desse referencial. Observe-se que o Metaio SDK permite obter não só os valores de translação e rotação associados a uma pose, como os seus inversos. Nas equações aqui apresentadas, as variáveis marcadas a negrito correspondem a vetores ou rotações.

Nesta abordagem, o referencial da marca i seria transladado segundo o vetor simétrico à translação da sua pose, para que a sua origem coincidisse com a origem do referencial da câmara, e depois segundo o vetor translação da pose da marca j , para que a sua origem se movesse da origem do referencial da câmara para a origem do referencial da marca j . Analogamente em termos de rotação, os eixos do referencial da

marca i seriam rodados de acordo com a rotação inversa à que transformou os eixos do referencial da câmara nos seus. Assim, os eixos do referencial da marca i passariam a coincidir com os do referencial da câmara. Em seguida, os eixos seriam rodados de acordo com a rotação que transformou os eixos do referencial da câmara nos eixos do referencial da marca j , logo passariam a coincidir com estes.

Esta abordagem ajuda a perceber o que se pretende fazer, mas não está correta. O problema é que as poses da marca i e j , tv_i e tv_j , são dadas relativamente ao referencial da câmara, logo as suas coordenadas, tanto para as translações como para os eixos das rotações, estão escritas na base desse referencial. A pose tv_k , porém, vai ser aplicada ao referencial da marca i e deve estar escrita na base desse mesmo referencial. É então preciso proceder a um processo de mudança de bases, que será explicado em seguida.

Denote-se por $(O_c, \mathbf{x}_c, \mathbf{y}_c, \mathbf{z}_c)$ o referencial da câmara, em que O_c é a origem do referencial e $\mathbf{x}_c, \mathbf{y}_c$ e \mathbf{z}_c são os vetores linearmente independentes que definem os seus eixos. Em particular, escritos na base do referencial da câmara, $\mathbf{x}_c = 1*\mathbf{x}_c + 0*\mathbf{y}_c + 0*\mathbf{z}_c = (1, 0, 0)$ e, analogamente, $\mathbf{y}_c = (0, 1, 0)$ e $\mathbf{z}_c = (0, 0, 1)$. Similarmente, denote-se por $(O_i, \mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$ o referencial da marca i e por $(O_j, \mathbf{x}_j, \mathbf{y}_j, \mathbf{z}_j)$ o referencial da marca j .

É preciso proceder a uma mudança de bases, do referencial da câmara para o referencial da marca i . Então, o primeiro passo consiste em escrever os vetores da base do referencial da câmara, $\mathbf{x}_c, \mathbf{y}_c$ e \mathbf{z}_c , em função (isto é, na base) dos vetores do referencial da marca i , $\mathbf{x}_i, \mathbf{y}_i$ e \mathbf{z}_i . A relação entre estes dois conjuntos de vetores é conhecida, sendo obtida a partir da rotação que transforma os eixos do referencial da câmara nos eixos do referencial da marca i , ou seja, a rotação da pose da marca i , $tv_i.rot$. Mais concretamente, deve-se rodar os eixos do referencial da marca i , escritos na base do mesmo, segundo $-tv_i.rot$, para que se transformem nos eixos do referencial da câmara e se consigam obter as coordenadas desses eixos na base do referencial da marca i . Com esta operação, obtém-se então:

$$\mathbf{x}_c = \mathbf{x}_i * (-tv_i.rot) = axx*\mathbf{x}_i + axy*\mathbf{y}_i + axz*\mathbf{z}_i$$

$$\mathbf{y}_c = \mathbf{y}_i * (-tv_i.rot) = ayx*\mathbf{x}_i + ayy*\mathbf{y}_i + ayz*\mathbf{z}_i$$

$$\mathbf{z}_c = \mathbf{z}_i * (-tv_i.rot) = azx*\mathbf{x}_i + azy*\mathbf{y}_i + azz*\mathbf{z}_i,$$

em que todos os coeficientes são conhecidos.

Além disto, graças a $-tv_i.trans$, que denota a translação que leva O_i para O_c e já está escrito na base do referencial da marca i , sabe-se que $O_c = (-tv_i.trans) + O_i$. Assim,

$$\begin{aligned} O_j &= tv_j.trans + O_c = \\ &= tv_j.trans.x*\mathbf{x}_c + tv_j.trans.y*\mathbf{y}_c + tv_j.trans.z*\mathbf{z}_c + O_c = \\ &= tv_j.trans.x*(axx*\mathbf{x}_i + axy*\mathbf{y}_i + axz*\mathbf{z}_i) + tv_j.trans.y*(ayx*\mathbf{x}_i + ayy*\mathbf{y}_i + ayz*\mathbf{z}_i) + \\ &tv_j.trans.z*(azx*\mathbf{x}_i + azy*\mathbf{y}_i + azz*\mathbf{z}_i) + O_c = \\ &= \dots = \\ &= \mathbf{b}.x*\mathbf{x}_i + \mathbf{b}.y*\mathbf{y}_i + \mathbf{b}.z*\mathbf{z}_i + O_c = \\ &= \mathbf{b}.x*\mathbf{x}_i + \mathbf{b}.y*\mathbf{y}_i + \mathbf{b}.z*\mathbf{z}_i + (-tv_i.trans) + O_i = \\ &= \mathbf{b} + (-tv_i.trans) + O_i, \end{aligned}$$

em que \mathbf{b} é um vetor obtido através dos cálculos denotados acima e corresponde ao vetor $tv_j.trans$ mas escrito na base do referencial da marca i , tal como $-tv_i.trans$.

Conseguiu-se então calcular a translação necessária para mover a origem do referencial da marca i para a origem do referencial da marca j , $tv_k.trans = (-tv_i.trans) + \mathbf{b}$.

Para calcular **tvk.rot**, um raciocínio semelhante é aplicado. Considere-se um ponto no espaço do referencial da marca j; para simplificar, tome-se o ponto P_{xj} , correspondente ao vértice do vetor \mathbf{x}_j quando este parte de O_j . Nos seguintes cálculos, identificam-se a castanho vetores/rotações escritos na base do referencial da marca j, a laranja vetores/rotações escritos na base do referencial da câmara e a verde pálido vetores/rotações escritos na base do referencial da marca i.

$$\begin{aligned}
 P_{xj} &= \mathbf{x}_j + O_j = \\
 &= \mathbf{xc} * \mathbf{tvj.rot} + O_j = \\
 &= \mathbf{xc} * \mathbf{tvj.rot} + \mathbf{tvj.trans} + O_c = \\
 &= \mathbf{xi} * (-\mathbf{tvi.rot}) * \mathbf{tvj.rot} + \mathbf{tvj.trans} + O_c = \\
 &= \dots = \\
 &= \mathbf{xi} * (-\mathbf{tvi.rot}) * \mathbf{tvj.rot} + \mathbf{b} + O_c = \\
 &= \mathbf{xi} * (-\mathbf{tvi.rot}) * \mathbf{tvj.rot} + \mathbf{b} + (-\mathbf{tvi.trans}) + O_i = \\
 &= \mathbf{xi} * (-\mathbf{tvi.rot}) * \mathbf{tvj.rot} + \mathbf{tvk.trans} + O_i.
 \end{aligned}$$

Os cálculos acima pretendem mostrar a relação entre um ponto escrito na base do referencial da marca j e um ponto escrito na base do referencial da marca i. Para se obterem as coordenadas do ponto na base do referencial da marca i, falta apenas converter uma rotação que ainda está escrita na base do referencial da câmara, **tvj.rot**. As rotações podem ser definidas por um eixo de rotação e um ângulo de rotação em torno desse eixo. Então, basta fazer uma mudança para a base i do vetor **tvj.rot.ax**, que representa o eixo da rotação que transforma os eixos do referencial da câmara nos eixos do referencial da marca j. Essa mudança de base é análoga à exemplificada no caso da translação e é a seguinte:

$$\begin{aligned}
 \mathbf{tvj.rot.ax} &= \mathbf{tvj.rot.ax.x} * \mathbf{xc} + \mathbf{tvj.rot.ax.y} * \mathbf{yc} + \mathbf{tvj.rot.ax.z} * \mathbf{zc} = \\
 &= \mathbf{tvj.rot.ax.x} * (\mathbf{axx} * \mathbf{xi} + \mathbf{axy} * \mathbf{yi} + \mathbf{axz} * \mathbf{zi}) + \mathbf{tvj.rot.ax.y} * (\mathbf{ayx} * \mathbf{xi} + \mathbf{ayy} * \mathbf{yi} + \mathbf{ayz} * \mathbf{zi}) + \\
 &\mathbf{tvj.rot.ax.z} * (\mathbf{azx} * \mathbf{xi} + \mathbf{azy} * \mathbf{yi} + \mathbf{azz} * \mathbf{zi}) = \\
 &= \dots = \\
 &= \mathbf{c.ax.x} * \mathbf{xi} + \mathbf{c.ax.y} * \mathbf{yi} + \mathbf{c.ax.z} * \mathbf{zi} = \\
 &= \mathbf{c.ax},
 \end{aligned}$$

em que **c.ax** é o eixo de uma rotação **c** que corresponde à rotação **tvj.rot** mas escrita na base do referencial da marca i.

Vem então que $P_{xj} = \mathbf{xi} * (-\mathbf{tvi.rot}) * \mathbf{tvj.rot} + \mathbf{tvk.trans} + O_i = \mathbf{xi} * (-\mathbf{tvi.rot}) * \mathbf{c} + \mathbf{tvk.trans} + O_i$, donde se conclui que a rotação necessária para rodar os eixos do referencial da marca i até coincidirem com os eixos do referencial da marca j é **tvk.rot** = $(-\mathbf{tvi.rot}) * \mathbf{c}$.

Anexo D: Estrutura do cérebro segundo vários atlas

Atlas Destrieux – 74 áreas por hemisfério

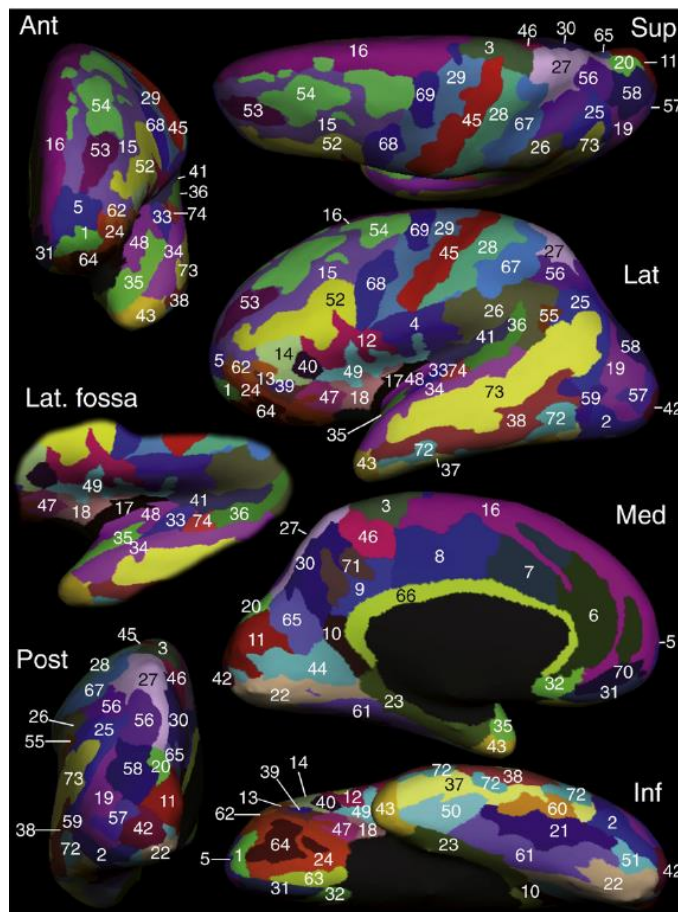


Figura D.1 – Parcelização de um hemisfério de acordo com o atlas Destrieux [44]

ID	Short Name	Long Name
1	G_and_S_frontomargin	Fronto-marginal gyrus (of Wernicke) and sulcus
2	G_and_S_occipital_inf	Inferior occipital gyrus (O3) and sulcus
3	G_and_S_paracentral	Paracentral lobule and sulcus
4	G_and_S_subcentral	Subcentral gyrus (central operculum) and sulci
5	G_and_S_transv_frontopol	Transverse frontopolar gyri and sulci
6	G_and_S_cingul-Ant	Anterior part of the cingulate gyrus and sulcus (ACC)
7	G_and_S_cingul-Mid-Ant	Middle-anterior part of the cingulate gyrus and sulcus (aMCC)
8	G_and_S_cingul-Mid-Post	Middle-posterior part of the cingulate gyrus and sulcus (pMCC)
9	G_cingul-Post-dorsal	Posterior-dorsal part of the cingulate gyrus (dPCC)
10	G_cingul-Post-ventral	Posterior-ventral part of the cingulate gyrus (vPCC, isthmus of the cingulate gyrus)
11	G_cuneus	Cuneus (O6)
12	G_front_inf-Opercular	Opercular part of the inferior frontal gyrus
13	G_front_inf-Orbital	Orbital part of the inferior frontal gyrus
14	G_front_inf-Triangul	Triangular part of the inferior frontal gyrus
15	G_front_middle	Middle frontal gyrus (F2)
16	G_front_sup	Superior frontal gyrus (F1)
17	G_Ins_lg_and_S_cent_ins	Long insular gyrus and central sulcus of the insula
18	G_insular_short	Short insular gyri

19	G_occipital_middle	Middle occipital gyrus (O2, lateral occipital gyrus)
20	G_occipital_sup	Superior occipital gyrus (O1)
21	G_oc-temp_lat-fusifor	Lateral occipito-temporal gyrus (fusiform gyrus, O4-T4)
22	G_oc-temp_med-Lingual	Lingual gyrus, ligual part of the medial occipito-temporal gyrus, (O5)
23	G_oc-temp_med-Parahip	Parahippocampal gyrus, parahippocampal part of the medial occipito-temporal gyrus, (T5)
24	G_orbital	Orbital gyri
25	G_pariet_inf-Angular	Angular gyrus
26	G_pariet_inf-Supramar	Supramarginal gyrus
27	G_parietal_sup	Superior parietal lobule (lateral part of P1)
28	G_postcentral	Postcentral gyrus
29	G_precentral	Precentral gyrus
30	G_precuneus	Precuneus (medial part of P1)
31	G_rectus	Straight gyrus, Gyrus rectus
32	G_subcallosal	Subcallosal area, subcallosal gyrus
33	G_temp_sup-G_T_transv	Anterior transverse temporal gyrus (of Heschl)
34	G_temp_sup-Lateral	Lateral aspect of the superior temporal gyrus
35	G_temp_sup-Plan_polar	Planum polare of the superior temporal gyrus
36	G_temp_sup-Plan_tempo	Planum temporale or temporal plane of the superior temporal gyrus
37	G_temporal_inf	Inferior temporal gyrus (T3)
38	G_temporal_middle	Middle temporal gyrus (T2)
39	Lat_Fis-ant-Horizont	Horizontal ramus of the anterior segment of the lateral sulcus (or fissure)
40	Lat_Fis-ant-Vertical	Vertical ramus of the anterior segment of the lateral sulcus (or fissure)
41	Lat_Fis-post	Posterior ramus (or segment) of the lateral sulcus (or fissure)
42	Pole_occipital	Occipital pole
43	Pole_temporal	Temporal pole
44	S_calcarine	Calcarine sulcus
45	S_central	Central sulcus (Rolando's fissure)
46	S_cingul-Marginalis	Marginal branch (or part) of the cingulate sulcus
47	S_circular_insula_ant	Anterior segment of the circular sulcus of the insula
48	S_circular_insula_inf	Inferior segment of the circular sulcus of the insula
49	S_circular_insula_sup	Superior segment of the circular sulcus of the insula
50	S_collat_transv_ant	Anterior transverse collateral sulcus
51	S_collat_transv_post	Posterior transverse collateral sulcus
52	S_front_inf	Inferior frontal sulcus
53	S_front_middle	Middle frontal sulcus
54	S_front_sup	Superior frontal sulcus
55	S_interm_prim-Jensen	Sulcus intermedius primus (of Jensen)
56	S_intrapariet_and_P_trans	Intraparietal sulcus (interparietal sulcus) and transverse parietal sulci
57	S_oc_middle_and_Lunatus	Middle occipital sulcus and lunatus sulcus
58	S_oc_sup_and_transversal	Superior occipital sulcus and transverse occipital sulcus
59	S_occipital_ant	Anterior occipital sulcus and preoccipital notch (temporo-occipital incisure)
60	S_oc-temp_lat	Lateral occipito-temporal sulcus
61	S_oc-temp_med_and_Lingual	Medial occipito-temporal sulcus (collateral sulcus) and lingual sulcus
62	S_orbital_lateral	Lateral orbital sulcus
63	S_orbital_med-olfact	Medial orbital sulcus (olfactory sulcus)
64	S_orbital-H_	Shaped Orbital sulci (H-shaped sulci)
65	S_parieto_occipital	Parieto-occipital sulcus (or fissure)
66	S_pericallosal	Pericallosal sulcus (S of corpus callosum)
67	S_postcentral	Postcentral sulcus

68	S_precentral-inf-part	Inferior part of the precentral sulcus
69	S_precentral-sup-part	Superior part of the precentral sulcus
70	S_suborbital	Suborbital sulcus (sulcus rostrales, supraorbital sulcus)
71	S_subparietal	Subparietal sulcus
72	S_temporal_inf	Inferior temporal sulcus
73	S_temporal_sup	Superior temporal sulcus (parallel sulcus)
74	S_temporal_transverse	Transverse temporal sulcus

Atlas Desikan-Killiany (DK) – 34 áreas por hemisfério

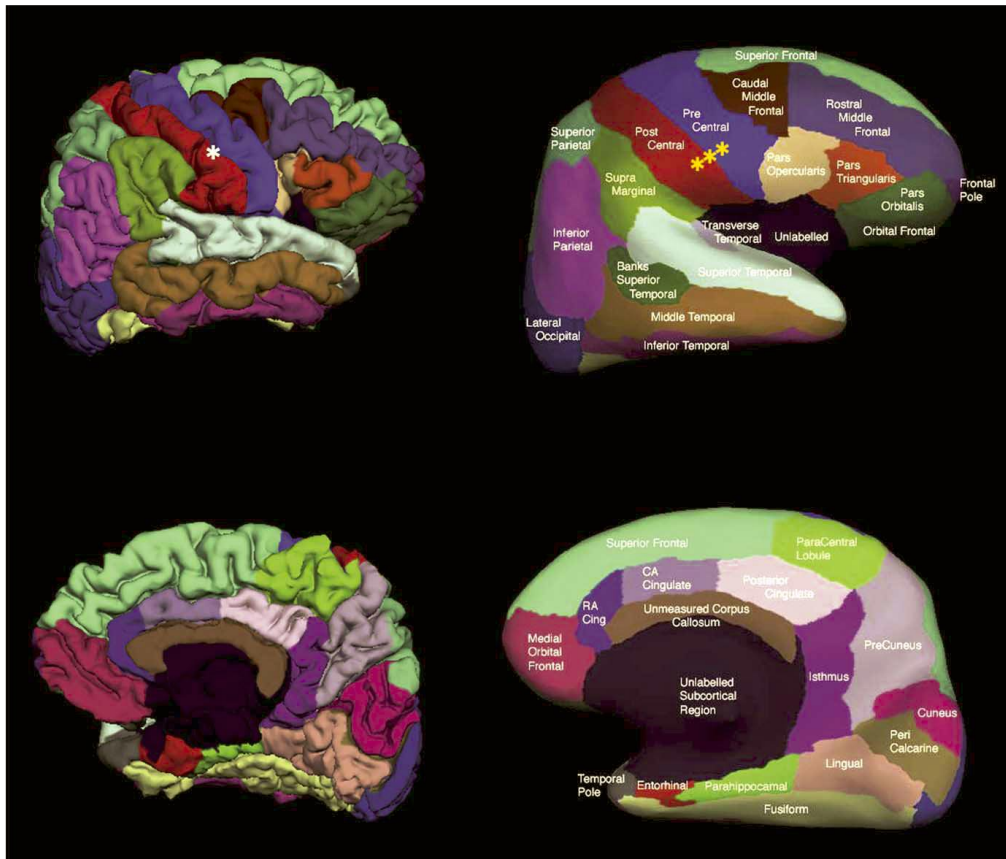


Figura D.2 – Parcelização de um hemisfério de acordo com o atlas Desikan-Killiany [45]

- Banks superior temporal sulcus
- Caudal anterior - cingulate cortex
- Caudal middle frontal gyrus
- Cuneus cortex
- Entorhinal cortex
- Frontal pole
- Fusiform gyrus
- Inferior parietal cortex
- Inferior temporal gyrus
- Insula
- Isthmus - cingulate cortex
- Lateral occipital cortex
- Lateral orbital frontal cortex
- Lingual gyrus
- Medial orbital frontal cortex
- Middle temporal gyrus
- Parahippocampal gyrus
- Paracentral lobule
- Pars opercularis
- Pars orbitalis
- Pars triangularis
- Pericalcarine cortex
- Postcentral gyrus
- Posterior - cingulate cortex
- Precentral gyrus
- Precuneus cortex
- Rostral anterior - cingulate cortex
- Rostral middle frontal gyrus
- Superior frontal gyrus
- Superior parietal cortex
- Superior temporal gyrus
- Supramarginal gyrus
- Temporal pole
- Transverse temporal cortex

Atlas Desikan-Killiany-Tourville (DKT) – 31 áreas por hemisfério

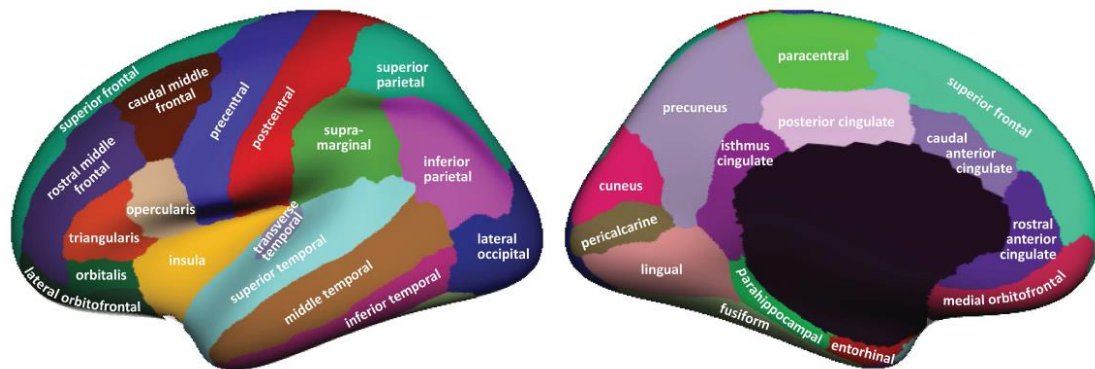


Figura D.3 – Parcelização de um hemisfério de acordo com o atlas Desikan-Killiany-Tourville [46]

- Caudal anterior - cingulate cortex
- Caudal middle frontal gyrus
- Cuneus cortex
- Entorhinal cortex
- Fusiform gyrus
- Inferior parietal cortex
- Inferior temporal gyrus
- Insula
- Isthmus - cingulate cortex
- Lateral occipital cortex
- Lateral orbital frontal cortex
- Lingual gyrus
- Medial orbital frontal cortex
- Middle temporal gyrus
- Parahippocampal gyrus
- Paracentral lobule
- Pars opercularis
- Pars orbitalis
- Pars triangularis
- Pericalcarine cortex
- Postcentral gyrus
- Posterior - cingulate cortex
- Precentral gyrus
- Precuneus cortex
- Rostral anterior - cingulate cortex
- Rostral middle frontal gyrus
- Superior frontal gyrus
- Superior parietal cortex
- Superior temporal gyrus
- Supramarginal gyrus
- Transverse temporal cortex

Anexo E: Divisão das áreas do atlas DK pelos 4 lobos

Lobo Frontal

- *Caudal anterior - cingulate cortex*
- Caudal middle frontal gyrus
- Frontal pole
- Lateral orbital frontal cortex
- Medial orbital frontal cortex
- Paracentral lobule
- Pars opercularis
- Pars orbitalis
- Pars triangularis
- Precentral gyrus
- *Rostral anterior - cingulate cortex*
- Rostral middle frontal gyrus
- Superior frontal gyrus

Lobo Occipital

- Cuneus cortex
- Lateral occipital cortex
- Lingual gyrus
- Pericalcarine cortex

Lobo Temporal

- Banks superior temporal sulcus
- Entorhinal cortex
- Fusiform gyrus
- Inferior temporal gyrus
- *Insula*
- Middle temporal gyrus
- Parahippocampal gyrus
- Superior temporal gyrus
- Temporal pole
- Transverse temporal cortex
- *Unknown*

Lobo Parietal

- Inferior parietal cortex
- *Isthmus - cingulate cortex*
- Postcentral gyrus
- *Posterior - cingulate cortex*
- Precuneus cortex
- Superior parietal cortex
- Supramarginal gyrus

Nota: As áreas indicadas a *itálico* levantam dúvidas sobre a sua pertença aos lobos, tendo sido lá colocadas por razões de simplificação da divisão.

Anexo F: Exemplo de Ficheiro de Definição das

Geometrias

70 35 35 26 20 14 10

white_DK_obj_dec/lh.white.DK.bankssts.obj L T Grey25
white_DK_obj_dec/lh.white.DK.caudalanteriorcingulate.obj L F Grey25
white_DK_obj_dec/lh.white.DK.caudalmiddlefrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.cuneus.obj L O Grey37,5
white_DK_obj_dec/lh.white.DK.entorhinal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.frontalpole.obj L F Grey50
white_DK_obj_dec/lh.white.DK.fusiform.obj L T Grey25
white_DK_obj_dec/lh.white.DK.inferiorparietal.obj L P Grey25
white_DK_obj_dec/lh.white.DK.inferiortemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.insula.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.isthmuscingulate.obj L P Grey50
white_DK_obj_dec/lh.white.DK.lateraloccipital.obj L O Grey50
white_DK_obj_dec/lh.white.DK.lateralorbitofrontal.obj L F Grey25
white_DK_obj_dec/lh.white.DK.lingual.obj L O Grey25
white_DK_obj_dec/lh.white.DK.medialorbitofrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.middletemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.paracentral.obj L F Grey50
white_DK_obj_dec/lh.white.DK.parahippocampal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.parsopercularis.obj L F Grey25
white_DK_obj_dec/lh.white.DK.parsorbitalis.obj L F Grey25
white_DK_obj_dec/lh.white.DK.parstriangularis.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.pericalcarine.obj L O Grey37,5
white_DK_obj_dec/lh.white.DK.postcentral.obj L P Red
white_DK_obj_dec/lh.white.DK.posteriorcingulate.obj L P Grey50
white_DK_obj_dec/lh.white.DK.precentral.obj L F Grey50
white_DK_obj_dec/lh.white.DK.precuneus.obj L P Grey25
white_DK_obj_dec/lh.white.DK.rostralanteriorcingulate.obj L F Grey25
white_DK_obj_dec/lh.white.DK.rostralmiddlefrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.superiorfrontal.obj L F Grey37,5
white_DK_obj_dec/lh.white.DK.superiorparietal.obj L P Grey50
white_DK_obj_dec/lh.white.DK.superiortemporal.obj L T Grey50
white_DK_obj_dec/lh.white.DK.supramarginal.obj L P Grey25
white_DK_obj_dec/lh.white.DK.temporalpole.obj L T Grey25
white_DK_obj_dec/lh.white.DK.transversetemporal.obj L T Grey37,5
white_DK_obj_dec/lh.white.DK.unknown.obj L O Grey37,5
white_DK_obj_dec/rh.white.DK.bankssts.obj R T Grey25
white_DK_obj_dec/rh.white.DK.caudalanteriorcingulate.obj R F Grey25
white_DK_obj_dec/rh.white.DK.caudalmiddlefrontal.obj R F Grey37,5
white_DK_obj_dec/rh.white.DK.cuneus.obj R O Grey37,5
white_DK_obj_dec/rh.white.DK.entorhinal.obj R T Grey50
white_DK_obj_dec/rh.white.DK.frontalpole.obj R F Grey50
white_DK_obj_dec/rh.white.DK.fusiform.obj R T Grey25
white_DK_obj_dec/rh.white.DK.inferiorparietal.obj R P Grey25
white_DK_obj_dec/rh.white.DK.inferiortemporal.obj R T Grey37,5
white_DK_obj_dec/rh.white.DK.insula.obj R T Grey37,5
white_DK_obj_dec/rh.white.DK.isthmuscingulate.obj R P Grey50
white_DK_obj_dec/rh.white.DK.lateraloccipital.obj R O Grey50
white_DK_obj_dec/rh.white.DK.lateralorbitofrontal.obj R F Grey25
white_DK_obj_dec/rh.white.DK.lingual.obj R O Grey25
white_DK_obj_dec/rh.white.DK.medialorbitofrontal.obj R F Grey37,5

white_DK_obj_dec/rh.white.DK.middletemporal.obj R T Grey37,5
white_DK_obj_dec/rh.white.DK.paracentral.obj R F Grey50
white_DK_obj_dec/rh.white.DK.parahippocampal.obj R T Grey50
white_DK_obj_dec/rh.white.DK.parsopercularis.obj R F Grey25
white_DK_obj_dec/rh.white.DK.parsorbitalis.obj R F Grey25
white_DK_obj_dec/rh.white.DK.parstriangularis.obj R F Grey37,5
white_DK_obj_dec/rh.white.DK.pericalcarine.obj R O Grey37,5
white_DK_obj_dec/rh.white.DK.postcentral.obj R P Red
white_DK_obj_dec/rh.white.DK.posteriorcingulate.obj R P Grey50
white_DK_obj_dec/rh.white.DK.precentral.obj R F Grey50
white_DK_obj_dec/rh.white.DK.precuneus.obj R P Grey25
white_DK_obj_dec/rh.white.DK.rostralanteriorcingulate.obj R F Grey25
white_DK_obj_dec/rh.white.DK.rostralmiddlefrontal.obj R F Grey37,5
white_DK_obj_dec/rh.white.DK.superiorfrontal.obj R F Grey37,5
white_DK_obj_dec/rh.white.DK.superiorparietal.obj R P Grey50
white_DK_obj_dec/rh.white.DK.superiortemporal.obj R T Grey50
white_DK_obj_dec/rh.white.DK.supramarginal.obj R P Grey25
white_DK_obj_dec/rh.white.DK.temporalpole.obj R T Grey25
white_DK_obj_dec/rh.white.DK.transversetemporal.obj R T Grey37,5
white_DK_obj_dec/rh.white.DK.unknown.obj R O Grey37,5

Anexo G: Script Python para decimar modelos no

Blender

```
import bpy

# Read file to get the list of models
file = open("C:\\Users\\José\\Documents\\Informática\\metaio\\metaio SDK 5.5.2\\_Android\\
Examples_SDK\\Example\\assets\\TutorialMyMetaioApp\\Assets\\
geometry_white_DK_dec_Grey_Marked.txt")

i = 0
for line in file:
    if i > 0:
        # Set Model
        filename = line.split(' ')[0]
        filename = filename[17:-4]

        # Import Model
        bpy.ops.import_scene.obj(filepath="C:\\Users\\José\\Documents\\Informática\\metaio\\metaio SDK
5.5.2\\_Android\\Examples_SDK\\Example\\assets\\
TutorialMyMetaioApp\\Assets\\white_DK_obj\\" + filename +
".obj")

        # Apply Decimate Modifier
        obj = bpy.data.objects[filename]
        bpy.context.scene.objects.active = obj

        bpy.ops.object.modifier_add(type='DECIMATE')
        obj.modifiers["Decimate"].ratio = 0.1

        for mod in obj.modifiers:
            if mod.type == "DECIMATE":
                bpy.ops.object.modifier_apply(apply_as='DATA', modifier=mod.name)

        # Export Model
        bpy.ops.export_scene.obj(filepath="C:\\Users\\José\\Documents\\Informática\\metaio\\metaio SDK
5.5.2\\_Android\\Examples_SDK\\Example\\assets\\
TutorialMyMetaioApp\\Assets\\white_DK_obj_dec\\" +
filename + ".obj", use_materials=False, axis_forward='-Z',
axis_up='-Y')

        # Delete Model From Scene
        bpy.ops.object.delete()

    i += 1

file.close()
```


Anexo H: Guião de Testes de Utilizador

Inquirição contextual para utilizadores da aplicação de Visualização do Cérebro em Realidade Aumentada e Virtual

Maio 2015

Introdução:

A Realidade Aumentada (RA) é uma técnica que permite sobrepor elementos virtuais a imagens reais, aumentando a informação real. Contrapõe-se à Realidade Virtual (RV), em que o utilizador percebe ambientes puramente virtuais.

A aplicação móvel que vai testar permite visualizar modelos virtuais tridimensionais do cérebro de pacientes, quer num contexto de RA – com o modelo sobreposto à cabeça do paciente –, quer num contexto de RV – em que o modelo se apresenta desligado do contexto real e o utilizador pode interagir com o mesmo. Os modelos tridimensionais são previamente obtidos e guardados no dispositivo móvel e o alinhamento dos mesmos com a cabeça do paciente no modo de RA é possibilitado pelo uso de marcas fixas numa touca que se coloca na cabeça do paciente. A aplicação permite ainda a visualização seletiva de áreas do cérebro.

A aplicação está a ser pensada para uso na técnica de Estimulação Magnética Transcraniana, em que um aparelho é usado para estimular determinadas áreas do cérebro, com o objetivo de curar certos problemas neuropsiquiátricos. É portanto importante para o médico conseguir visualizar exatamente onde se encontra a área a estimular, daí a utilidade desta aplicação.

Dados pessoais:

Idade: _____

Sexo:

Masculino	Feminino
-----------	----------

Habilitações Literárias: _____

Área de formação (se superior): _____

Ocupação: _____

Experiência de utilização de *smartphones*:

1 (pouca)	2	3	4	5 (muita)
-----------	---	---	---	-----------

Experiência de utilização de ecrãs táteis:

1 (pouca)	2	3	4	5 (muita)
-----------	---	---	---	-----------

Conhecimentos de Realidade Aumentada ou Virtual:

1 (poucos)	2	3	4	5 (muitos)
------------	---	---	---	------------

Observações adicionais:

--

Guião de utilização:

1º - Iniciar a aplicação.

2º - Calibrar as marcas, guiando-se pelas indicações no ecrã. Para tal, apontar com a câmara ao mesmo tempo para uma marca não calibrada e uma marca calibrada (inicialmente apenas a marca 5 está calibrada). Quando todas as marcas estiverem calibradas uma indicação aparecerá e depois o modelo do cérebro ficará visível (desde que se esteja a apontar para alguma das marcas) e os indicadores de calibração desaparecerão.

a) Facilidade do processo de calibração:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

b) Duração do processo de calibração:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

c) Utilidade dos indicadores de calibração:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Tem alguma sugestão em relação aos indicadores de calibração (ex. texto ou imagem, cor, posição, ...)?

--

e) Observações sobre o processo de calibração:

--

3º - Faça uma rotação completa em torno do paciente mantendo a câmara do dispositivo sempre a apontar para a cabeça deste.

Se eventualmente encontrar alguma marca mal calibrada (isto é, se ao apontar exclusivamente para essa marca o modelo virtual do cérebro aparecer numa posição/orientação completamente errada) então retire-a da touca.

a) Teve de retirar alguma marca da touca? Número de marcas retiradas: _____

b) Conseguiu visualizar o modelo do cérebro a partir de todos os ângulos que percorreu?

Sim	Não
-----	-----

c) Qualidade do alinhamento do modelo virtual:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Tem alguma sugestão em relação à representação/visualização do cérebro?

--

e) Observações/sugestões:

--

4º - Utilize a régua como se fosse o aparelho de estimulação e simule a estimulação da área do cérebro assinalada a vermelho, aproximando o aparelho dessa área (nota: a marca presente na régua tem de ser captada pela câmara). Repare como a cor do aparelho de estimulação muda conforme a distância do aparelho à cabeça do paciente.

a) Facilidade de encontrar a área assinalada a vermelho:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

b) Perceção de profundidade:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

c) Utilidade da indicação colorida da distância no aparelho de estimulação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Intuitividade do código de cores usado para representar a distância:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

e) Qualidade da representação visual do aparelho de estimulação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

f) Qualidade da experiência de estimulação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

g) Observações/sugestões:

--

5º - Carregue no botão “Options” para entrar no menu de opções (caso os botões estejam invisíveis toque no ecrã). Neste menu, na secção “Stimulation Device Glyph”, selecione a opção “Coil”, para alterar o glifo usado na representação do aparelho de estimulação. Para confirmar e voltar ao ecrã de Realidade Aumentada, prima “OK” e depois simule novamente a estimulação da área assinalada a vermelho.

a) Perceção de profundidade:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

b) Qualidade da representação visual do aparelho de estimulação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

c) Qualidade da experiência de estimulação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Apreciação global do modo de Realidade Aumentada:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

e) Observações/sugestões:

--

6º - Carregue no botão “VR” para entrar no modo de interação em Realidade Virtual (caso os botões estejam invisíveis toque no ecrã).

- 1. Ajuste o tamanho do modelo virtual colocando dois dedos no ecrã e afastando-os/aproximando-os.**
- 2. Rode o modelo virtual colocando dois dedos no ecrã e rodando-os relativamente um ao outro.**
- 3. Altere o eixo de rotação do modelo virtual carregando no botão “Horiz.” (caso os botões estejam invisíveis toque no ecrã) e rode novamente o modelo.**
- 4. Pense numa perspetiva a partir da qual seria interessante visualizar o modelo do cérebro. De seguida, rode o modelo na aplicação até o visualizar dessa perspetiva.**
- 5. Carregue nos botões do lado esquerdo da interface da aplicação para obter vistas predefinidas (caso os botões estejam invisíveis toque no ecrã).**

a) Facilidade de ajustar o tamanho do modelo:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

b) Intuitividade do gesto para ajustar o tamanho do modelo:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

c) Facilidade de colocar o modelo numa orientação desejada sem recorrer às vistas predefinidas:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Intuitividade do gesto para rodar o modelo:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

e) Utilidade das vistas predefinidas:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

f) Apreciação global do modo de Realidade Virtual:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

g) Observações/sugestões:

--

7º - Carregue no botão “Options” para entrar no menu de opções (caso os botões estejam invisíveis toque no ecrã). Neste menu, na secção “Regions to Visualize”, verifique que todas as caixas de verificação se encontram marcadas.

- 1. Desmarque o lobo frontal e o lobo occipital. Observe como algumas das caixas de verificação (relativas às áreas do cérebro pertencentes a esses dois lobos) ficaram desmarcadas. Prima “OK” para confirmar e voltar ao ecrã de Realidade Virtual. Verifique que apenas são representadas as áreas do cérebro pertencentes aos lobos parietal e temporal.**
- 2. Prima “AR” para entrar no modo de Realidade Aumentada e, apontando a câmara para a cabeça do paciente, verifique que continuam a ser representadas apenas as áreas seleccionadas no menu de opções.**
- 3. Volte a entrar no menu de opções e desmarque o hemisfério esquerdo. Confirme a escolha e verifique que as áreas do hemisfério esquerdo que ainda estavam visíveis também desapareceram. Observe como existe uma área assinalada a verde que não era visível inicialmente, por ser interior ao cérebro.**
- 4. Entre no modo de Realidade Virtual e volte ao menu de opções. Marque/desmarque individualmente algumas das áreas à sua escolha. Confirme a escolha e verifique o resultado.**

a) Organização do menu de opções:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

b) Facilidade de alterar as áreas visíveis:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

c) Utilidade da funcionalidade de visualização seletiva:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Apreciação global da funcionalidade de visualização seletiva:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

e) Observações/sugestões:

--

8º - Termine a aplicação.

a) Forma usada para terminar a aplicação: _____

9º - Tarefa final: Entre na aplicação, calibre as marcas, visualize a área assinalada a verde e simule a estimulação nessa área, usando a régua.

a) Tempo usado para calibrar as marcas: _____

b) Tempo usado para visualizar a área assinalada a verde: _____

c) Apreciação global das funcionalidades da aplicação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

d) Observações/sugestões acerca das funcionalidades da aplicação:

--

e) Apreciação global das capacidades de visualização dadas pela aplicação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

f) Observações/sugestões acerca das capacidades de visualização dadas pela aplicação:

--

g) Apreciação global da interface de utilizador da aplicação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

h) Observações/sugestões acerca da interface de utilizador da aplicação:

--

i) Apreciação global da utilidade/relevância da aplicação:

1 (péssima)	2	3	4	5 (excelente)
-------------	---	---	---	---------------

Obrigado!

Anexo I: Fotografias dos Testes de Utilizador



Figura I.1 – Fotografias dos testes de utilizador realizados

Bibliografia

- [1] *BioISI | Faculdade de Ciências da Universidade de Lisboa* (último acesso em junho de 2015), <http://www.fc.ul.pt/en/unidade/bioisi>
- [2] *Clínica Neurológica e Estimulação Magnética Transcraniana* (último acesso em abril de 2015),
<http://www.estimulacaoneurologica.com.br/estimulacao.aspx>
- [3] *Android Developer Tools | Android Developers* (último acesso em junho de 2015),
<http://developer.android.com/tools/help/adt.html>
- [4] *Eclipse - The Eclipse Foundation open source community website.* (último acesso em junho de 2015), <https://eclipse.org/>
- [5] *metaio | home | Augmented Reality Products & Solutions* (último acesso em junho de 2015), <http://www.metaio.com/>
- [6] *Blender.org – Home of the Blender project – Free and Open 3D Creation Software* (último acesso em junho de 2015), <http://www.blender.org/>
- [7] José Soeiro, Ana Paula Cláudio, Beatriz Carmo, Hugo Alexandre Ferreira, *Visualizing the Brain on a Mixed Reality Smartphone Application*, Artigo aceite para a conferência EMBC 2015, 37ª Conferência Internacional Anual da “IEE Engineering in Medicine and Biology Society”, Milão, Itália, 25 a 29 de Agosto de 2015
- [8] *EMBC15* (último acesso em junho de 2015), <http://embc.embs.org/2015/>
- [9] *Iterative development model – Iterative and incremental development – Wikipedia, the free encyclopedia* (ultimo acesso em junho de 2015), https://en.wikipedia.org/wiki/Iterative_and_incremental_development#/media/File:Iterative_development_model.svg
- [10] *Mixed reality – Wikipedia, the free encyclopedia* (último acesso em junho de 2015), https://en.wikipedia.org/wiki/Mixed_reality
- [11] *Reality-virtuality continuum – Wikipedia, the free encyclopedia* (último acesso em maio de 2015), https://en.wikipedia.org/wiki/Reality%E2%80%93virtuality_continuum
- [12] *Augmented reality – Wikipedia, the free encyclopedia* (último acesso em junho de 2015), https://en.wikipedia.org/wiki/Augmented_reality

- [13] *Untitled Document* (último acesso em junho de 2015), <http://sweb.cityu.edu.hk/sm2203/oldclass2007/lab1/toc02/index.html>
- [14] *Virtual reality – Wikipedia, the free encyclopedia* (último acesso em junho de 2015), https://en.wikipedia.org/?title=Virtual_reality
- [15] *Transcranial magnetic stimulation – Wikipedia, the free encyclopedia* (último acesso em junho de 2015), https://en.wikipedia.org/wiki/Transcranial_magnetic_stimulation
- [16] *Schizophrenia – Wikipedia, the free encyclopedia* (último acesso em junho de 2015), https://en.wikipedia.org/wiki/Schizophrenia#Positive_and_negative
- [17] *Tablet ajuda na cirurgia de fígado* (último acesso em novembro de 2014), <http://www.alexandreporfirio.com/2013/08/20/noticias/tablet-ajuda-na-cirurgia-de-figado/>
- [18] S. M. Romano, *Realidade Aumentada Aplicada à Medicina*
- [19] *Óculos revelam tecido canceroso em tempo real* (último acesso em novembro de 2014), http://www.dn.pt/inicio/ciencia/interior.aspx?content_id=3681152
- [20] *Google Glass proof of concept* (último acesso em novembro de 2014), <http://www.youtube.com/watch?v=ssldTFWBv3E>
- [21] *Philips Sees Medtech Disruption in Google Glass | Qmed* (último acesso em junho de 2015), <http://www.qmed.com/mpmn/article/philips-sees-medtech-disruption-google-glass>
- [22] *Digital Health Intelligence :: Google Glass delivers patient data* (último acesso em junho de 2015), <http://www.digitalhealth.net/news/ehi/8959/google-glass-delivers-patient-data/>
- [23] *Aplicações de Realidade Aumentada* (último acesso em novembro de 2014), <http://www.ckirner.com/download/tutoriais/rv-sibgrapi97/aplic.htm>
- [24] *The Future of Medical Visualisation*, MIT – Technology Review: The Future of Medical Visualisation
- [25] V. Ferrari, G. Megali, E. Troia, A. Pietrabissa, and F. Mosca, *A 3-D mixed-reality system for stereoscopic visualization of medical dataset*, IEEE Trans Biomed Eng, vol. 56, pp. 2627-33, 2009.
- [26] K. Konishi, M. Hashizume, M. Nakamoto, Y. Kakeji, I. Yoshino, A. Taketomi, Y. Sato, S. Tamura, Y. Maehara, *Augmented reality navigation system for endoscopic*

surgery based on three-dimensional ultrasound and computed tomography: Application to 20 clinical cases, CARS, 2005

- [27] *Realidade aumentada permite ver o próprio cérebro em ação* (último acesso em novembro de 2014), http://www.swissinfo.ch/por/internacional_afp/Realidade_aumentada_permite_ver_proprio_cerebro_em_acao.html?cid=38577520
- [28] G. Echegaray, I. Herrera, I. Aguinaga, C. Buchart, D. Borro, *A Brain Surgery Simulator*, IEEE Computer Graphics and Applications, pp. 12-18, 2014
- [29] F. Cutolo, P. D. Parchi, V. Ferrari, *Video See Through AR Head-Mounted Display for Medical Procedures*, IEEE International Symposium on Mixed and Augmented Reality 2014, Science and Technology Proceedings, Munique, Alemanha, 10 a 12 de Setembro de 2014
- [30] D. Katic, A. Wekerle, J. Gortler, P. Spengler, S. Bodenstedt, S. Rohl, S. Suwelack, H. G. Kenngott, M. Wagner, B. P. Muller-Stich, R. Dillman, S. Speidel, *Context-aware Augmented Reality in laparoscopic surgery*, Computerized Medical Imaging and Graphics, vol. 37, pp. 174-182, 2013
- [31] C. A. Linte, K. P. Davenport, K. Cleary, C. Peters, K. G. Vosburgh, N. Navab, P. Edwards, P. Janin, T. M. Peters, D. R. Holmes III, R. A. Robb, *On mixed reality environments for minimally invasive therapy guidance: Systems architecture, successes and challenges in their implementation from laboratory to clinic*, Computerized Medical Imaging and Graphics, vol. 37, pp. 83-97, 2013
- [32] *Samsung GALAXY S4 - Life companion* (último acesso em novembro de 2014), <http://www.samsung.com/global/microsite/galaxys4/>
- [33] *Durovis Dive – 3D Virtual Reality Gaming on a Smartphone* (último acesso em abril de 2015), <https://www.durovis.com/index.html>
- [34] *Vuforia Augmented Reality for 3D Mobile Content | Qualcomm Vuforia* (último acesso em novembro de 2014), <https://www.qualcomm.com/products/vuforia>
- [35] *ID Marker | metaio Developer Portal* (último acesso em junho de 2015), <http://dev.metaio.com/sdk/documentation/tracking-config/optical-tracking/id-marker/>
- [36] *Quaternions and spatial rotation – Wikipedia, the free encyclopedia* (último acesso em março de 2015), https://en.wikipedia.org/wiki/Quaternions_and_spatial_rotation

- [37] *Taxicab geometry* – *Wikipedia, the free encyclopedia* (último acesso em março de 2015), https://en.wikipedia.org/wiki/Taxicab_geometry
- [38] *Brain for Blender | Brainder*. (último acesso em junho de 2015), <http://brainder.org/download/brain-for-blender/>
- [39] C. P. Halsted, *Brightness, luminance and confusion*, Information display, 1993
- [40] J. L. Gabbard, J. E. Swan II, D. Hix, *The Effects of Text Drawing Styles, Background Textures, and Natural Lighting on Text Legibility in Outdoor Augmented Reality*, Presence, vol. 15, Fevereiro 2006
- [41] R. Ribeiro, *Desenvolvimento de uma interface de visualização de conectividade cerebral usando realidade virtual e controlo por gestos*, Tese de Mestrado, FCT-UNL, 2014
- [42] *FreeSurfer* (último acesso em junho de 2015), <http://freesurfer.net/>
- [43] *MATLAB – The Language of Technical Computing* (último acesso em junho de 2015), <http://www.mathworks.com/products/matlab/>
- [44] C. Destrieux, B. Fischl, A. Dale, E. Halgren, *Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature*, NeuroImage, vol. 53, pp. 1-15, 2010
- [45] R. S. Desikan, F. Ségonne, B. Fischl, B. T. Quinn, B. C. Dickerson, D. Blacker, R. L. Buckner, A. M. Dale, R. P. Maguire, B. T. Hyman, M. S. Albert, R. J. Killiany, *An automated labelling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest*, NeuroImage, vol. 31, pp. 968-980, 2006
- [46] A. Klein, J. Tourville, *101 labeled brain images and a consistent human cortical labeling protocol*, Frontiers in Neuroscience, vol. 6, Dezembro 2012
- [47] *CorticalParcellation – Free Surfer Wiki* (último acesso em junho de 2015), <https://surfer.nmr.mgh.harvard.edu/fswiki/CorticalParcellation>
- [48] C. Stephani, G. Fernandez-Baca Vaca, R. Maciunas, M. Koubeissi, H. O. Lüders, *Functional neuroanatomy of the insular lobe*, Brain Struct Funct, vol. 216, pp. 137-149, 2011
- [49] *Blender Documentation Contents – Blender 2.75.1 – API documentation* (último acesso em junho de 2015), http://www.blender.org/api/blender_python_api_2_75_1/
- [50] *ORA by Optinvent* (último acesso em junho de 2015), <http://optinvent.com/see-through-glasses-ORA>

