



UNIVERSIDADE DA BEIRA INTERIOR
Engenharia

Interaction in Games Playing with Move motion controller

Mauro Filipe Martins Ferrão

Dissertação para obtenção do grau de Mestre em
Engenharia Informática
(2º ciclo de estudos)

Orientador: Prof. Doutor Frutuoso Silva

Covilhã, 27 de Junho de 2011

Agradecimentos

Quero dedicar este espaço a todos os que, directa ou indirectamente, foram importantes para mim na elaboração deste trabalho.

Não querendo desprestigiar ninguém, porque todos foram importantes, quero no entanto ressaltar alguns, como por exemplo o meu orientador, o Prof. Doutor Frutuoso G. M. Silva, pela forma como me conduziu ao longo de todo este processo. A sua disponibilidade, a sua boa vontade e a sua orientação foram premissas importantes para a concretização deste.

Gostaria também de agradecer a toda a equipa docente do Departamento de Informática da Universidade da Beira Interior, bem como aos meus colegas que foram importantes com a sua colaboração. Quero ainda agradecer à professora Kelly O'Hara do Departamento de Ciências do Desporto da UBI e à professora Marta André da Escola Secundária/3 Quinta das Palmeiras pela disponibilidade demonstrada na realização dos testes que envolveram os seus alunos.

À minha família, que esteve sempre do meu lado, como a minha mãe Ana Maria que disse "presente" em todos os momentos em que eu precisava, bem como aos meus amigos, especialmente à Margarida Pereira que me ajudou em alguns passos da criação deste projecto, e a todos os que contribuíram jogando o jogo mSquash o que permitiu recolher dados importantes para analisar, quero agradecer profundamente por todo o apoio dado.

A todos os que não foram referidos, mas que não são menos importantes, o meu muito obrigado.

Resumo

As técnicas de interacção humana com o computador (HCI) têm evoluído muito nos últimos anos, o que trouxe benefícios para a indústria dos videojogos. O aparecimento de comandos de movimento como o comando Wii Remote ou o comando Playstation Move são exemplos desta evolução. Uma área dos videojogos que beneficiou bastante com esta evolução foi a área dos jogos que promovem o exercício físico, os Exergames.

Esta dissertação apresenta o jogo mSquash, um Exergame que foi criado para testar a potencialidade deste tipo de videojogos e também das técnicas de interacção usadas neles. Para isso foi usado o comando de movimento Playstation Move como controlador do jogo e o motor de renderização OGRE3D como plataforma base para o desenvolvimento do jogo. Foram ainda efectuados testes com vários jogadores e foram retirados dados relativos à sua performance no jogo. Os jogadores foram ainda sujeitos a um questionário, depois de terem jogado o mSquash, onde se tentou perceber a sua opinião sobre o jogo e a forma de interacção utilizada.

Palavras chave

Exergames, videojogos, comando Move, serious games, interacção gestual

Abstract

Human-Computer Interaction (HCI) techniques have evolved over the last years, which brings benefits to the video game industries. The appearances of motion controllers like the Playstation Move and the Wii Remote are examples of this evolution. One type of video games that benefits a lot from this evolution were the Exergames, video games that promote physical exercise.

This dissertation presents the mSquash, an Exergame that was created to study the capabilities of this type of video games and the interaction techniques used on them. Playstation Move was used as the controller for this game, which was built using the 3D rendering engine OGRE. Besides were realized tests with several players and were generated data about the players performance in the game. The players also had to answer to a questionnaire, after playing the game, to obtain their opinions about the game and about the interaction method used.

Keywords

Exergames, video games, Move controller, serious games, motion tracking

Conteúdo

Agradecimentos	iii
Resumo	v
Abstract	vii
1 Introdução	1
1.1 Introdução	1
1.2 Objectivos da dissertação	3
1.3 Estrutura da dissertação	3
2 Trabalhos relacionados	5
2.1 Dispositivos de interacção em videojogos	6
2.2 HCI: Interacção Homem-Computador	11
2.3 Exergames	13
2.4 OGRE: Exemplo de jogos	25
3 Trabalho desenvolvido	29
3.1 Ferramentas usadas	29
3.1.1 OGRE	29
3.1.2 FMOD	31
3.1.3 OpenCV	33
3.1.4 Periféricos	33
3.2 O jogo mSquash	37

3.2.1	Regras de jogo	38
3.2.1.1	Sistema de pontuação	41
3.2.1.2	Relatório de sessão	43
3.3	mSquash: Estrutura do código	43
3.4	Motor de física	45
3.5	Tracking	50
4	Testes e resultados	55
4.1	Testes	55
4.2	Resultados	57
5	Conclusões e trabalho futuro	71
Anexo:	Instalação do jogo mSquash	81
Instalação do comando Move		81
Instalação da câmara PSEye		83
Configuração do jogo		85

Lista de Figuras

2.1	Controladores do jogo Pong, versão arcade[5] (esquerda) e versão caseira (direita)[3]	6
2.2	Atari Football e o controlador trackpad(esquerda) [1] e o jogo Pole Position com o respectivo volante[2]	7
2.3	O primeiro joystick para videojogos (esquerda) e um joystick actual (direita)	7
2.4	Gamepad da consola Nintendo NES	8
2.5	Alguns dos gamepads mais importantes: a)Nintendo SNES b)Sega Mega Drive c)Microsoft Xbox360 d)Playstation 3 e)Nintendo Wii	8
2.6	Novos controladores de movimento: Wii Remote (cima à esquerda), Move (cima à direita) e Kinect (baixo)	9
2.7	Consola portátil 3DS da Nintendo	10
2.8	Ilustração 9 - Jogador a usar uma interface BCI para jogar World of Warcraft (imagem retirada de [8])	10
2.9	Esquema do HCI aplicado aos videojogos (baseado em [66])	11
2.10	Esquema dos 5 componentes dos Exergames (imagem retirada de [52])	14
2.11	Modelo Dual Flow (Retirado de [50])	15
2.12	Exergame de uma maratona virtual implementado em [67]	17
2.13	Esquema dos 5 componentes de um Exergame com a introdução de um "corpo virtual"(retirado de [52])	17
2.14	Individuo a jogar Astrojumper (Imagem retirada de [17])	18
2.15	Jogadora a realizar testes numa bicicleta adaptada a um Exergame (Imagem retirada de [51])	19

2.16	Exemplo do jogo Easybowling com a máquina (esquerda) e a simulação da colisão (direita) (Imagem retirada de [43])	20
2.17	Joyboard (esquerda) e o jogo Mogul Maniac (direita)(Imagem retirada de [7])	20
2.18	Acessório Foot Craz (Imagem retirada de[7])	21
2.19	Nintendo Power Pad	21
2.20	DDR versão Arcade (esquerda) e o controlador para a versão Playstation (direita)	22
2.21	Exemplo da jogabilidade do jogo EyeToy:Kinetic	23
2.22	Wii Balance Board	23
2.23	Ecrã de jogo Wii Fit Plus	24
2.24	Exemplo do jogo EA Sports Active 2[14]	24
2.25	Screenshot do jogo Anhk	25
2.26	Screenshot do jogo Earth eternal	26
2.27	Screenshot do jogo Garshasp	26
2.28	Screenshot do jogo Nimbus	27
2.29	Screenshot do jogo Torchlight	27
3.1	Principais objectos do OGRE(retirado de [59])	31
3.2	Playstation Eye	34
3.3	Playstation Move	35
3.4	Screenshot do jogo Under Siege	36
3.5	Vários acessórios para o comando Playstation Move	37
3.6	Jogo de squash(Imagem retirada de [4])	38
3.7	Modo de jogo Treino	39
3.8	Modo de jogo Normal	40
3.9	Modo de jogo Target	40
3.10	Instruções da pontuação no modo Normal	41
3.11	Instruções da pontuação no modo Target	42
3.12	Exemplo dos vários níveis de dificuldade no jogo	42
3.13	Esquema do código do jogo mSquash	43

3.14	Representação da posição da bola	46
3.15	Representação das velocidades da bola	47
3.16	Esquema da colisão da bola com a parede	49
3.17	Captura original da câmara	50
3.18	Imagem suavizada	51
3.19	Imagem após o filtro de Threshold	51
3.20	Imagem após todos os filtros	52
3.21	Calculo do diâmetro e dos valores máximos de X e de Y	53
3.22	Calculo da posição central da bola	53
4.1	Sessão de jogo na Escola Quinta das Palmeiras	56
4.2	Gráfico da distribuição da população de acordo com o género	58
4.3	Gráfico da distribuição da população de acordo com a idade	58
4.4	Gráfico da pontuação média final de acordo com o género do jogador	59
4.5	Gráfico da pontuação média em cada bola por nível de dificuldade	59
4.6	Gráfico da pontuação final média por idade	60
4.7	Gráfico do tempo de jogo médio gasto por sessão	61
4.8	Gráfico do número de bolas pontuadas por zona	61
4.9	Gráfico do número de bolas perdidas por zona	62
4.10	Gráfico da pontuação final média por PC	63
4.11	Gráfico da pontuação final média por tipo de jogador	63
4.12	Gráfico da pontuação final média relacionado com o hábito de fazer desporto	64
4.13	Gráfico da pontuação final média relacionado com o grau de dificuldade atribuído	65
4.14	Gráfico da pontuação final média relacionado com o hábito de jogar com comandos de movimento	65
4.15	Gráfico da pontuação final média relacionado com o hábito de jogar desportos com raquete	66
4.16	Gráfico da pontuação atribuída ao mSquash em relação ao benefício para o bem-estar e saúde dos jogadores	68

1	Carregar drivers na ferramenta DS3 Tool	82
2	Conectar o comando por Bluetooth	83
3	Configurar as opções da câmara	84
4	Ajuste dos parâmetros da câmara	84
5	Espaço de jogo correctamente configurado	86
6	Imagem correctamente configurada (esquerda) e incorrectamente con- figurada (direita)	86
7	Introdução dos dados do utilizador	87
8	Ecrã de selecção do tipo de jogo	88
9	Jogo a correr	88

Lista de Tabelas

3.1	Especificações da câmara PSEye[55]	34
3.2	Especificações do comando Playstation Move[58]	35
4.1	Pontuações do jogo atribuídas nos questionários	67
4.2	Características dos jogadores retiradas das respostas dos questionários	67
4.3	Aspectos positivos do jogo mSquash apontados pelos jogadores testados	68
4.4	Aspectos negativos do jogo mSquash apontados pelos jogadores testados	69

Capítulo 1

Introdução

1.1 Introdução

Os videojogos são, cada vez mais, uma das principais formas de entretenimento na sociedade actual. Isto pelo modo como permite uma maior interacção multimédia, bem como pelos desafios que propõe ao jogador, aliando ainda, os benefícios que podem trazer à saúde e bem-estar do mesmo.

Segundo um estudo recente [50], as crianças dos 8 aos 18 anos passam mais tempo em frente das televisões, computadores e ecrãs de jogos (44.5 horas semanais) do que em qualquer outra actividade nas suas vidas, excepto dormir. Segundo [45] o homem adulto joga em média 7.6 horas por semana de jogos digitais enquanto as mulheres da mesma faixa etária jogam cerca de 7.4 horas. As crianças dos 8 aos 10 anos são quem joga mais, gastando diariamente 1 hora e 33 minutos em média em videojogos, e com uma probabilidade de 73% de jogar pelo menos uma vez por dia. É por isso cada vez mais importante olhar para os videojogos como uma ferramenta preciosa para alterar certos comportamentos prejudiciais da nossa sociedade e não só como sendo uma das causas.

Com a evolução da tecnologia, foram introduzidos novos dispositivos no nosso dia-a-dia que visam facilitar as nossas vidas. Os telemóveis, por exemplo, estão já introduzidos de um modo tão natural que chega a parecer ridículo alguém não possuir um destes aparelhos. O grande problema é que facilitando as tarefas diárias, mesmo que seja algo útil, acaba por contribuir para um estilo de vida mais sedentário. Não é por isso de estranhar que a sociedade tenha ficado mais rica a nível tecnológico e

intelectual mas com um grande custo, toda esta evolução também traz consigo diversos problemas de saúde, tal como a obesidade.

A Organização Mundial da Saúde registou no ano de 2009 que 60% da população Portuguesa vivia em áreas urbanas e que no ano de 2008 mais de metade da população tinha excesso de peso e cerca de 21% sofria de obesidade[19]. Para combater estes factores é importante a criação de várias soluções que possam ajudar na saúde da população. Tendo em conta o tempo gasto pelos jovens e adultos com os videojogos e o défice de saúde provocado pela inactividade física, é preciso arranjar uma solução que consiga ligar estes dois factores, e é neste âmbito que surgem os videojogos que requerem exercício físico, também designados de Exergames.

Os Exergames como foram apelidados, são jogos que usam o exercício físico como forma de interacção. Para além de oferecerem uma experiência de jogo mais divertida e se tornarem mais atractivos para todos os tipos de idades, podem também ter um papel importante na introdução de uma vida saudável na população jovem ou de ajudar a manter a população idosa mais activa.

Uma das maiores, senão a maior característica dos Exergames é a interacção entre o jogador e o jogo. Apesar da Realidade Virtual já ser uma tecnologia com vários anos, nunca teve grande sucesso comercial na área dos Exergames, talvez pelos preços elevados ou mesmo pela falta de qualidade das versões "caseiras" deste equipamento. Uma das excepções foi a série de jogos Dance Dance Revolution[22], que consistia num jogo com ritmo musical em que o jogador tinha de coordenar os movimentos com a música usando um tapete como controlador.

No ano de 2006 surge então uma revolução nesta área, com o lançamento da consola Wii[38] por parte da Nintendo, consola que surgiu com uma nova mentalidade, usando um comando de movimento para permitir novas experiências de jogo, mais centradas na interacção entre o jogador e videojogo. As consolas "rivais", até então quase só preocupadas em proporcionar uma maior experiência visual e sonora, perceberam ao ver o sucesso da Wii que a aposta em novas tecnologias de interacção seria uma escolha acertada. Foi sem espanto por isso que a Sony Playstation lançou o comando de movimento Move[58] e a Microsoft a tecnologia Kinect[32]. Com a atenção das grandes empresas de videojogos focadas nos jogos e na interacção baseada no movimento do jogador, começaram a surgir diversos jogos que obrigam ao exercício físico (Exergames).

É importante por isso estudar e avaliar as diferentes formas de interacção existentes

(e futuras) nos videojogos, focando especialmente os Exergames.

1.2 Objectivos da dissertação

O objectivo principal desta dissertação foi o planeamento e desenvolvimento de um videojogo que pudesse ser usado para avaliar a interacção usando o comando Move. Para isso foi criado um jogo de computador baseado no squash, com uma forte componente física usando o comando Move da Playstation, podendo este ser uma ferramenta importante para a obtenção de resultados que permitissem tirar conclusões tanto sobre interacção homem-computador (HCI) como sobre os jogos que envolvem exercício físico (Exergames).

1.3 Estrutura da dissertação

Esta dissertação está estruturada do seguinte modo:

- No capítulo 1 é feita uma introdução à dissertação, é explicado o objectivo desta e é descrita a estrutura da dissertação;
- No capítulo 2 são apresentadas as tecnologias existentes nas áreas dos videojogos, HCI e Exergames, a sua importância e os trabalhos realizados nestas áreas;
- No capítulo 3 é explicado o trabalho desenvolvido para a resolução do objectivo proposto, sendo neste caso a criação do jogo mSquash e a tecnologia usada na criação deste;
- No capítulo 4 são descritos os testes efectuados com um grupo de utilizadores e são apresentados os principais resultados obtidos para os diversos utilizadores nas várias sessões realizadas com o jogo mSquash.
- No capítulo 5 são apresentadas as conclusões e é apresentado o trabalho futuro planeado para este projecto.

Capítulo 2

Trabalhos relacionados

Os videojogos são actualmente umas das principais formas de entretenimento. Estes podem ser considerados uma junção da tecnologia computacional com os jogos tradicionais e outras formas de arte, como os filmes, a música, etc. Surgiram inicialmente nos salões de jogos e foram depois evoluindo para versões caseiras, com o seu aparecimento nos computadores pessoais e com o nascimento das consolas de videojogos.

Com o avanço da tecnologia e com o aumento de popularidade dos videojogos, o seu objectivo deixou de ser apenas o entretenimento puro. Passaram a ser criados jogos que pudessem oferecer algo mais às pessoas do que apenas uma experiência de diversão.

Os serious games, como o seu nome indica, são videojogos que têm objectivos sérios, como o ensino, o treino, etc. Estes podem ser usados para simular de modo realista várias situações úteis para adquirir experiência em diversas áreas, poupando dinheiro e garantindo a segurança dos utilizadores. Este tipo de jogos, por exemplo, vão desde simuladores de voo a jogos de treino militar (America's Army[63]), passando pelos jogos educacionais. Dentro deste tipo de jogos encontramos também os Exergames, jogos que promovem o exercício físico dos jogadores, que se focam na saúde dos mesmos, e que actualmente estão muito em voga devido especialmente aos comandos que permitem mapear os movimentos do utilizador (ex: Wii remote, Move ou Kinect).

O avanço da tecnologia permitiu também o aumento do realismo deste tipo de jogos, através de melhores modelos 3D, de uma melhoria na inteligência artificial, de novas formas de interacção com o jogo, entre outras características. Tudo isto permite uma maior imersão na simulação e consequentemente uma maior aproximação à realidade.

2.1 Dispositivos de interacção em videojogos

Quando surgiram, os primeiros videojogos usavam hardware especializado que ia além das economias dos jogadores caseiros. Aparecerem em primeiro lugar nos salões de jogos, e eram jogados em máquinas especiais para cada tipo de jogo.

Pong, lançado em 1972 para os salões de jogos pela Atari, foi o primeiro videojogo a ter sucesso comercial, e foi o impulsionador para o começo da indústria dos videojogos[61]. É um jogo para dois jogadores que simula o ténis de mesa e usa controladores rotativos analógicos (ver Figura 2.1). Em 1975 foi lançada a versão caseira do Pong, também com os controladores rotativos.



Figura 2.1: Controladores do jogo Pong, versão arcade[5] (esquerda) e versão caseira (direita)[3]

Em 1978, a Atari lança o Atari Football, um jogo de futebol americano controlado por uma trackball (ver Figura 2.2), um controlador que consiste numa bola que se pode rodar em qualquer direcção para controlar o jogo. Outro tipo de controlador que surgiu nos salões de jogo e que se tornou popular foi o volante, usado em simuladores de corridas (ver Figura 2.2).

Os joysticks (ver Figura 2.3) foram também tipos de controladores importantes para os videojogos nos seus anos iniciais. Já usados em aviação e em máquinas industriais, foram adaptados aos videojogos em 1967 por Ralph H. Baer [15]. São constituídos por uma base e por um manipululo (stick) vertical que ao ser inclinado para um dos lados transmite a informação desse lado ao sistema. Com o passar dos anos foram-lhes incluídos vários melhoramentos como mais botões ou mesmo tecnologia háptica para simular forças.



Figura 2.2: Atari Football e o controlador trackpad(esquerda) [1] e o jogo Pole Position com o respectivo volante[2]



Figura 2.3: O primeiro joystick para videojogos (esquerda) e um joystick actual (direita)

Como até então cada tipo de controlador era criado para um jogo específico, isso criava um problema para o mercado, não era viável a oferta de diversos controladores por parte das empresas de videojogos. Era preciso por isso encontrar um controlador genérico. Foi então que a Nintendo lançou o primeiro "gamepad" em 1983, para a consola NES (Nintendo Entertainment System) [46]. Este controlador (ver Figura 2.4) tem um conjunto de botões direccionais em cruz (D-Pad) para usar com o polegar esquerdo, dois botões vermelhos (rotulados A e B) para usar com a mão direita e ainda os botões centrais "Select" e "Start", originalmente criados para controlar as opções, começar e pausar o jogo.



Figura 2.4: Gamepad da consola Nintendo NES

O sucesso deste tipo de controlador levou a que fossem criados diversos periféricos baseados neste sistema (ver Figura 2.5). Com a evolução dos videojogos, e o aumento de complexidade e sofisticação, foi necessário evoluir também os controladores, de modo a que possam permitir uma melhor interacção entre o jogador e o jogo.



Figura 2.5: Alguns dos gamepads mais importantes: a)Nintendo SNES b)Sega Mega Drive c)Microsoft Xbox360 d)Playstation 3 e)Nintendo Wii

Com essa evolução os gamepads tornaram-se mais ergonómicos e foram-lhes adicionadas novas funcionalidades como a inclusão de mais botões, de joysticks, de sensores de pressão nos botões, sensores de movimento e também lhes foram removidas

outras características menos vantajosas como a sua conexão por fio, passando a ser usadas tecnologias sem fios (como a tecnologia Bluetooth) para comunicar com as consolas.

Um exemplo dessa evolução é o gamepad da Sony Playstation mais recente, denominado DualShock3 [54]. É constituído por 2 joysticks, 17 botões (com 12 a medirem a pressão exercida), sensores de movimento, bateria incorporada e tecnologia sem fios Bluetooth.

Para contrariar esta complexidade cada vez maior dos controladores de videojogos, foram sendo criados novos métodos de controlo, como os sensores de movimento ou captura e processamento de imagem (tracking).

O lançamento da Nintendo Wii e o seu controlador Wii Remote (ver Figura 2.6) em 2006 foi um dos primeiros passos para introduzir um novo tipo de interacção nos videojogos, o movimento. Este controlador contém um acelerómetro que permite detectar movimento em 3 dimensões e uma câmara infra-vermelhos que capta a sua posição. Após o sucesso deste controlador as grandes empresas de videojogos concorrentes lançaram também acessórios para as suas consolas que permitissem o controlo do jogo através de movimento: O Move por parte da Sony Playstation e o Kinect por parte da Microsoft (ver figura2.6).



Figura 2.6: Novos controladores de movimento: Wii Remote (cima à esquerda), Move (cima à direita) e Kinect (baixo)

Com a contínua evolução da tecnologia, iremos assistir ao aparecimento de novos métodos de interacção com os videojogos. Exemplo disso é a tecnologia 3D estereoscópica, que está a ganhar cada vez mais importância no mundo dos videojogos (e dos filmes). Esta transforma imagens bidimensionais em imagens que permitem dar a ilusão de terem 3 dimensões. A nova consola portátil da Nintendo, a Nintendo 3DS [37] (ver Figura 2.7), já permite visualizar os jogos em 3 dimensões sem ser necessário o uso de óculos especiais. Esta consola também possui várias câmaras para tirar partido da realidade aumentada.



Figura 2.7: Consola portátil 3DS da Nintendo

Outro exemplo de uma tecnologia que pode vir a ser revolucionária foi apresentada no estudo realizado por Bos et al. [8], em que foram feitos testes jogando vários jogos usando interfaces controladas pelo cérebro (Brain-Computer Interfaces) (ver Figura 2.8).



Figura 2.8: Ilustração 9 – Jogador a usar uma interface BCI para jogar World of Warcraft (imagem retirada de [8])

2.2 HCI: Interação Homem-Computador

De acordo com o designer de jogos Chris Crawford[11], o aspecto gráfico, a animação, o som e a música são elementos essenciais de um videojogo mas não são o principal elemento, sendo este a interação. Em [53], Sinha et al. definem a Human Computer Interaction (HCI) como sendo uma disciplina preocupada com o design, avaliação e implementação de sistemas de computação interactivos para uso humano e também do estudo dos fenómenos que o rodeiam. Neste artigo são também referenciados os termos que devem ser considerados na implementação de um sistema HCI: a funcionalidade, que é definida pelas acções ou serviços que fornece aos utilizadores e a usabilidade, que define a capacidade do sistema em ser usado para cumprir certos objectivos de modo eficiente e adequado, tendo em conta cada utilizador.

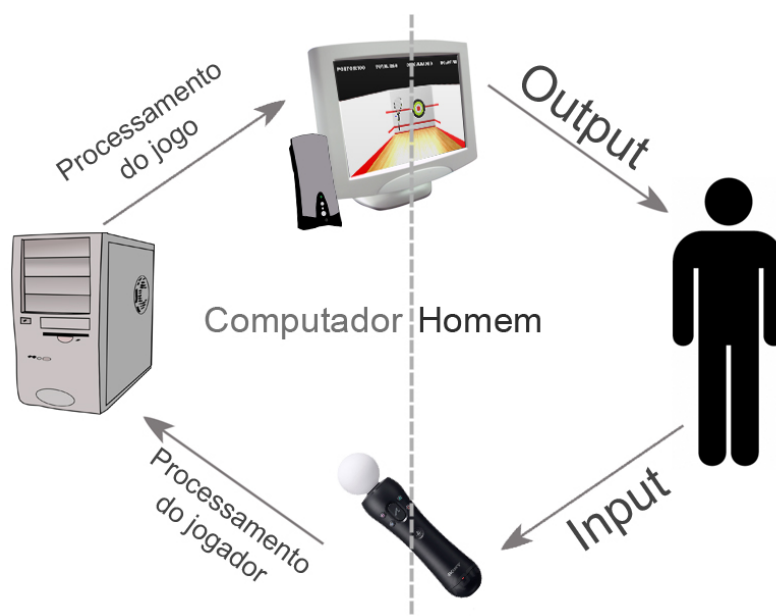


Figura 2.9: Esquema do HCI aplicado aos videojogos (baseado em [66])

Segundo Zhang et al. [66], a interação homem-computador (HCI) aplicada aos videojogos consiste num ciclo com 4 passos (ver Figura 2.9). O computador começa por apresentar o ambiente virtual ao jogador através de imagens, som ou outros modos de percepção. A este passo foi dado o nome de Output. Baseando-se na informação recebida, o jogador decide o que tem a fazer e usa os controladores do jogo para efectuar uma acção (o Input). O comando fornece os dados do jogador e envia-os para o computador. Foi chamado a esse passo o Processamento do Jogador. No último passo,

a que damos o nome de Processamento do Jogo, o computador processa as acções no jogo que são afectadas pelos Input do jogador e calcula também as suas pontuações, segundo as regras de jogo, enviando depois essas informações para os dispositivos de saída. Depois deste último passo volta ao primeiro passo e começa mais uma ronda do ciclo da HCI.

Em [9], Xinyuan baseia-se nas experiências e teorias criadas no campo da HCI para ajudar no planeamento e criação de um videojogo. Para isso devemos aplicar o princípio "Simples, natural, amigável e consistente" à parte da interacção homem-computador do jogo.

O princípio da simplicidade afirma que ao processo de interacção entre o homem e o computador deve ser o mais simples possível. Segundo este princípio os meios de interacção demasiado complicados podem dificultar a vida ao jogador e interromper a experiência de jogo, que acaba por reduzir o efeito de entretenimento do jogo. Os designers de jogos devem focar-se essencialmente na experiência de jogo em vez de se focarem no processo de interacção. Tendo isto em conta, devem ser os jogos a apresentar o desafio ao jogador, não os controladores de jogo.

Segundo o princípio da naturalidade, o processo de HCI no jogo deve ter em conta a experiência de vida e os hábitos do jogador. Ou seja, a jogabilidade deve permitir ao jogador que execute as acções no jogo de um modo mais natural possível, sem que ele tenha que pensar cada vez que tiver que pressionar um botão, por exemplo. Em relação ao Output, o jogo deve ter certas semelhanças com a realidade, para que os jogadores tenham maior facilidade em aceitar o mundo virtual criado, dando uma maior sensação de imersão na experiência.

O sistema deve ser também amigável para o jogador, dando a informação necessária ao jogador para este perceber e interagir correctamente com o jogo. A informação visual deve ser agrupada logicamente e disponibilizada ao jogador de um modo perceptível, de modo a que este perceba quais as informações mais importantes e as que podem ser ignoradas. O jogo também deve fornecer ajudas ao jogador quando este se estiver a desviar do caminho pretendido pelo jogo, só quando necessário para não estragar a experiência de jogo, como no caso dos jogos em que o jogador deve explorar o cenário ou aprender através do método tentativa erro. O jogador deve ser sempre informado do estado do jogo para poder tomar decisões de acordo com as circunstâncias, como por exemplo um jogador que é avisado que a sua personagem do videojogo está a perder vida por estar em contacto com um certo objecto. Recebendo essa informação vai tentar

agir de modo diferente para não prejudicar a sua personagem virtual. Por fim, o jogo deve responder às acções do jogador (feedback), de modo a que este perceba que as realizou, como a vibração de um comando quando ocorre uma colisão no jogo ou um som a avisar que o jogador teve sucesso numa dada tarefa.

O princípio da consistência afirma que o Output do computador e o Input do jogador devem manter a consistência na aparência e na lógica. Esta consistência mantém a coerência da experiência de jogo e permite ao jogador que se habitue ao jogo e que possa melhorar a sua performance. No caso de haver uma eventual falta de consistência, iria prejudicar toda a experiência de jogo, porque o jogador não saberia exactamente como proceder.

2.3 Exergames

Como vimos na introdução desta dissertação, uma das soluções para alguns dos problemas de saúde da sociedade consiste numa maior aposta nos Exergames. Estes jogos podem encorajar ao exercício tornando a actividade física mais interessante para os jogadores com um estilo de vida mais sedentário, ou seja, os mais propensos a ter problemas de saúde.

Segundo Sinclair et al.[52], os Exergames são constituídos por 5 componentes base: o software de jogo, os aparelhos de feedback (vídeo e som), o controlador do Exergame, os sensores de monitorização de sinais vitais e o componente considerado mais importante, o jogador(ver Figura 2.10).

Os controladores de jogo de um Exergame enviam sinais para a aplicação de acordo com os movimentos do jogador, substituindo assim os comandos normais por algo que obrigue o jogador a gastar energia.

A aplicação envia a resposta (feedback) através dos aparelhos de feedback (ecrãs, colunas, etc.) e também através de sinais hápticos (vibração, força, etc.) enviados pelos controladores de jogo.

Para além de interagir com a aplicação usando os controladores de movimento, o jogador também envia os seus dados vitais pelos biosensores. Fornecendo assim o seu estado fisiológico para que a aplicação possa garantir que o jogador esteja a fazer o esforço pretendido e também que o faça em segurança.

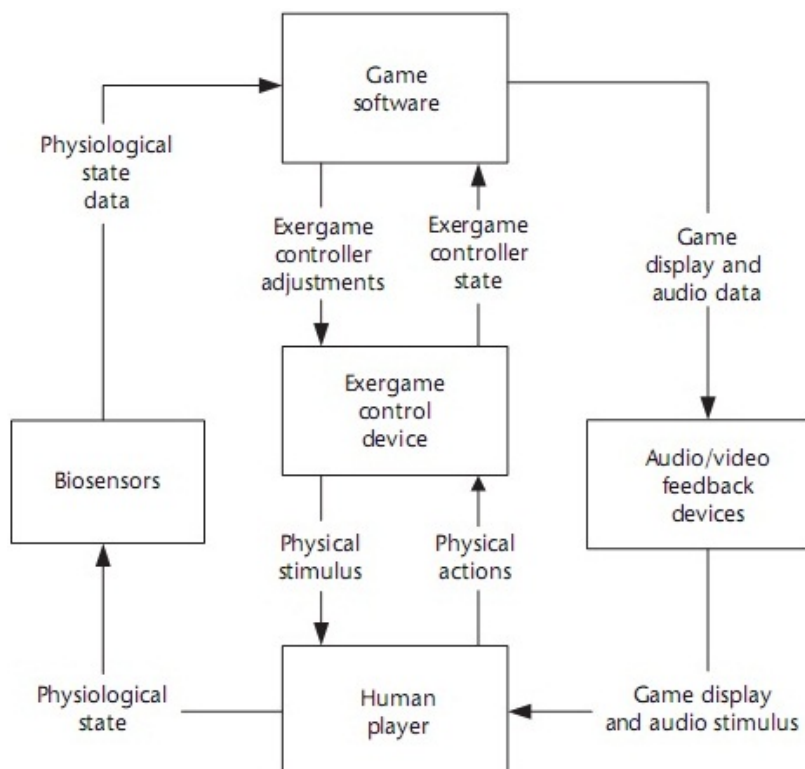


Figura 2.10: Esquema dos 5 componentes dos Exergames (imagem retirada de [52])

Em [50] é apresentado o conceito de fluxo ("flow") apresentado inicialmente por Csikszentmihalyi em 1975 [12] mas adaptado para videogames. Fluxo é o estado de empenhamento numa actividade, em que a pessoa está totalmente concentrada. Na versão modificada de fluxo temos os seguintes componentes:

- **Concentração:** os jogos requerem concentração por parte do jogador, e este deve poder-se concentrar nele.
- **Desafio:** os jogos devem ser suficientemente desafiantes e igualar o nível de habilidade do jogador.
- **Controlo:** os jogadores devem sentir uma sensação de controlo sobre as suas acções no jogo.
- **Habilidade do jogador:** os jogos devem suportar o desenvolvimento e domínio das habilidades do jogador.
- **Objectivos claros:** os objectivos devem ser apresentados de forma clara e no tempo apropriado.

- Feedback: os jogadores devem receber uma resposta adequada em tempo próprio para isso.
- Imersão: os jogadores devem experimentar um envolvimento profundo no jogo.
- Interacção social: os jogos devem suportar e criar oportunidades para interacção social.

Também em [50], foi criada uma escala para avaliar os valores de atractividade e eficácia dos Exergames, o modelo Dual Flow. Segundo o gráfico da Figura 2.11, a parte da atractividade mostra-nos que o ideal é haver um equilíbrio entre o desafio (challenge) e a habilidade (skill), se o desafio for grande e a habilidade for pouca chegamos a um estado de ansiedade. No caso de a habilidade ser elevada e o desafio ser pequeno é provável que o jogador fique aborrecido com o jogo. Se as duas componentes forem baixas gera-se um estado de apatia. No gráfico da eficácia verificamos que se a intensidade (intensity) for alta e o nível físico do utilizador (fitness) baixo resulta em fracasso, se o inverso suceder há deterioração, e caso os dois parâmetros sejam baixos verificamos que não há benefício no Exergame.

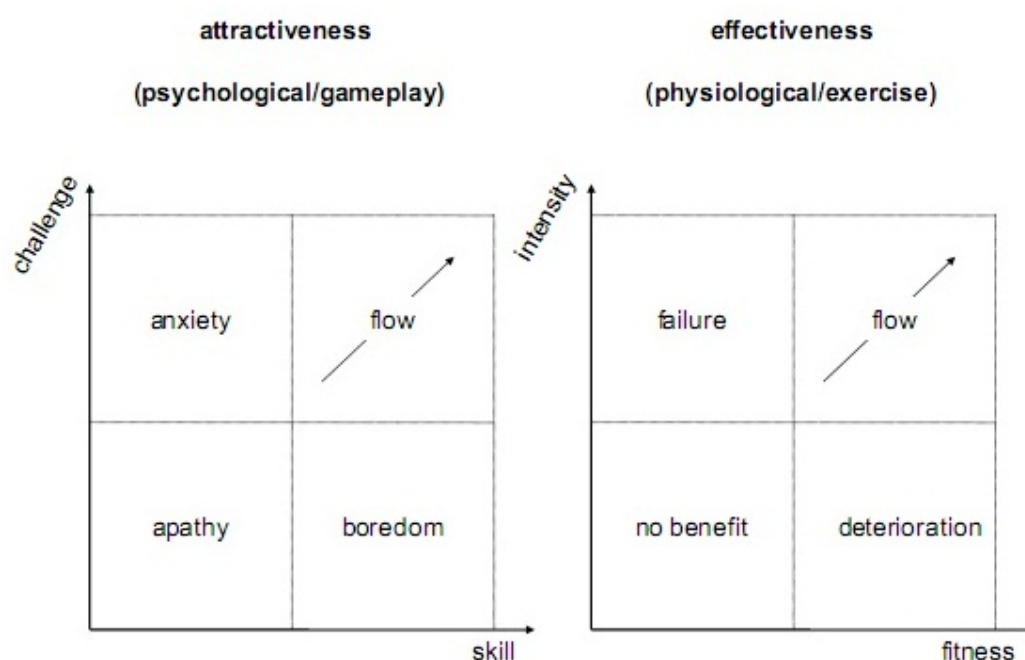


Figura 2.11: Modelo Dual Flow (Retirado de [50])

Segundo Andrew Macvean [25], os Exergames têm essencialmente dois tipos de objectivos:

- Objectivos a curto prazo: consistem em encorajar o jogador a fazer exercício físico durante o videojogo
- Objectivos a longo Prazo: pretendem alterar a atitude do jogador perante o exercício físico em geral, levando a que estes o pratiquem mesmo fora do âmbito dos Exergames.

O mesmo autor afirma que existem dois contextos abordados pelos Exergames, o contexto envolvendo o Ego do jogador (Ego-Involved), em que são usados objectivos que quando completados aumentam o ego do indivíduo, como recompensas ou aplausos. O outro contexto envolve tarefas (Task-Involved) em que o jogador tenta ter sucesso apenas com o interesse de melhorar a sua capacidade física. O problema da primeira abordagem é que só motiva o jogador enquanto o jogo decorre, ou caso tenha más prestações, só o desmotiva (a curto prazo). No segundo caso o jogador fica com vontade de melhorar as suas capacidades, expandindo-se essa ideia para a sua vida diária e não apenas para o jogo. Segundo este artigo, a segunda abordagem é a que merece mais foco por parte dos criadores de jogo, e a que pode ser mais benéfica para os jogadores.

No artigo [67] é apresentado o modelo ISCAL(Immersion, Scientificalness, Competitiveness, Adaptability and Learning) adaptado a um jogo que consiste numa maratona virtual (ver Figura 2.12). Segundo este artigo, o modelo contém as características chave que devem estar presentes num Exergame, e são estas:

- Imersão: sensação do jogador estar a participar num desporto real, e depende da qualidade do ambiente virtual, como os gráficos, o som e o feedback dos controladores.
- Rigor científico: consiste em executar o exercício físico de modo a garantir a sua eficácia evitando excessos que podem ser prejudiciais.
- Adaptabilidade: característica que define o ajuste da dificuldade por parte do jogo de modo a manter o jogador na zona entre muito difícil e demasiado fácil.
- Competitividade: capacidade do jogo em criar um desafio para o jogador e o recompensar caso jogue bem.
- Aprendizagem: capacidade do jogo de incitar novos conhecimentos ao jogador, adquiridos pela experiência de jogo.

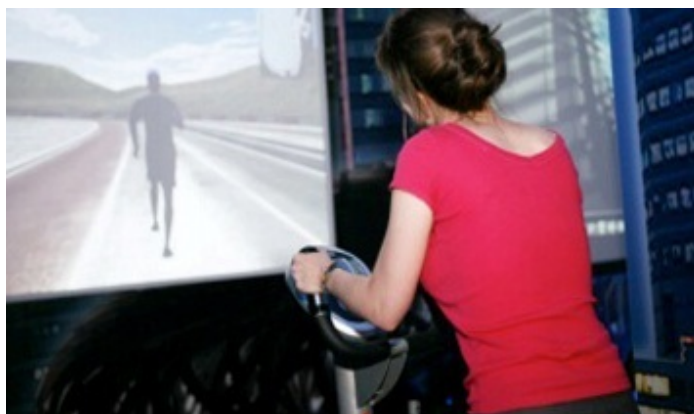


Figura 2.12: Exergame de uma maratona virtual implementado em [67]

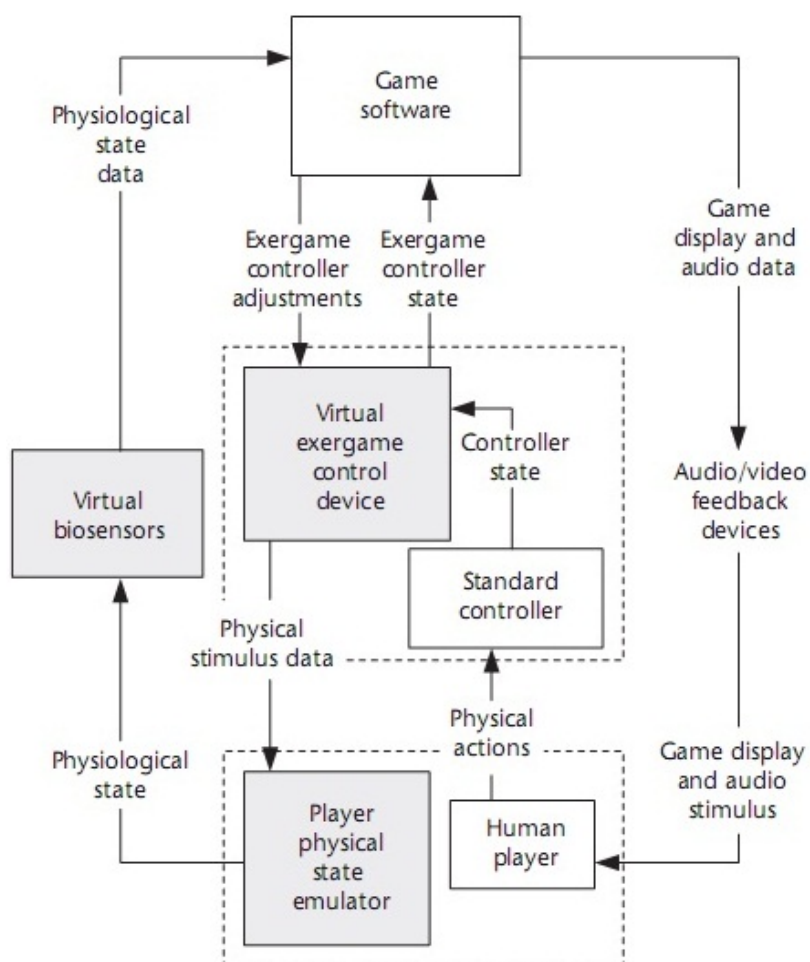


Figura 2.13: Esquema dos 5 componentes de um Exergame com a introdução de um "corpo virtual"(retirado de [52])

Tal como é preciso testar, melhorar e ajustar um videogame "normal", também na

criação de um Exergame é preciso realizar diversos testes, mas mais direccionados para a parte física, pretendendo testar a segurança e eficácia do jogo.

Em [52], Sinclair et al. criaram uma ferramenta que permite testar os Exergames e simular o efeito destes na pessoa testada, implementando para isso um "corpo virtual". Este serve para remover e emular a parte física dos testes e assim diminuir os custos e tempo gasto no processo, comparativamente aos testes feitos com pessoas. A imagem da figura 2.13 mostra o esquema apresentado na figura 2.10 alterado para introduzir esta ferramenta de simulação.

No artigo [6] foram feitos testes a 39 crianças com médias de idade de 11 anos para avaliar os gastos energéticos de diversos Exergames comparando-os com os gastos de andar ou correr numa passadeira. A unidade de medida destes gastos foi o equivalente metabólico, ou MET. Segundo o estudo, andar a aproximadamente 5 Km/h na passadeira gasta 4.9 METs nas crianças. Dos 6 jogos avaliados 5 tiveram um gasto energético superior ao valor da passadeira, mostrando que os Exergames conseguem atingir o seu objectivo, de promover a actividade física nos jogadores.

Foi apresentado, no artigo [17], um Exergame para motivar jogadores com necessidades especiais à prática de actividade física. AstroJumper é um Exergame criado especialmente tendo em conta as necessidades das crianças com autismo. Este jogo foi instalado numa cave com 3 projectores estereoscópicos permitindo assim visão 3D usando óculos polarizados. O jogo tem como ambiente o espaço e é pedido ao jogador para se desviar de vários objectos que voam em direcção ao jogador, e ao mesmo tempo tentar agarrar outro tipo de objectos para aumentar a sua pontuação (ver Figura 2.14).



Figura 2.14: Indivíduo a jogar Astrojumper (Imagem retirada de [17])

Na dissertação de Johanna Hoysniemi [21], a autora examinou o entretenimento, usabilidade e utilidade dos Exergames usando vários casos de estudo:

- Um jogo infantil (QuiQui's Giant Bounce[20]) controlado por movimento e som usando uma câmara Web e um microfone.
- Um jogo de artes marciais (Kick Ass Kung-Fu) usando uma vista de perfil e dois ecrãs, um de cada lado do jogador.
- Um jogo de dança (Dance Dance Revolution [22])

Em [51] foi usada uma GameBike da marca Cateye modificada para jogar um Exergame de plataformas em que se controla um helicóptero (ver Figura 2.15). A altura do helicóptero é controlada pela intensidade da rotação dada aos pedais da bicicleta. Este teste pretendia testar a eficácia e atractividade de um Exergame (seguindo o modelo Dual Flow já apresentado), medindo e registando os valores dos batimentos cardíacos dos jogadores durante a sessão de jogo, as pontuações de jogo e as respostas dadas em questionários realizados após cada sessão.

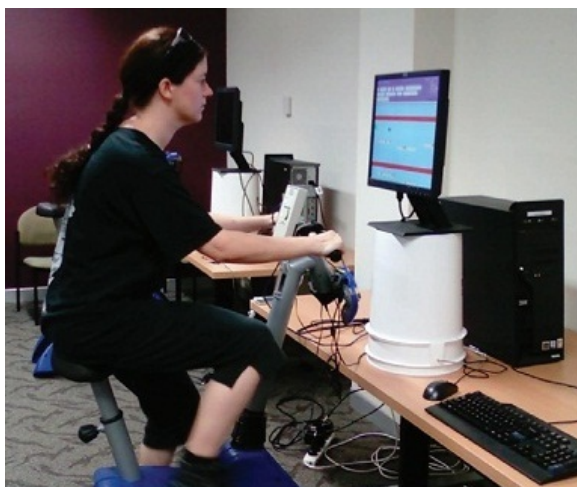


Figura 2.15: Jogadora a realizar testes numa bicicleta adaptada a um Exergame (Imagem retirada de [51])

Apesar de Exergame ser uma definição relativamente recente, já existiam jogos criados antes dessa mesma definição que promoviam o desporto e a actividade física. Um exemplo disso mesmo é o projecto Easybowling em 2002 [43], um projecto em que foi criada uma máquina para simular o Bowling usando alguns componentes da realidade virtual(ver Figura 2.16).

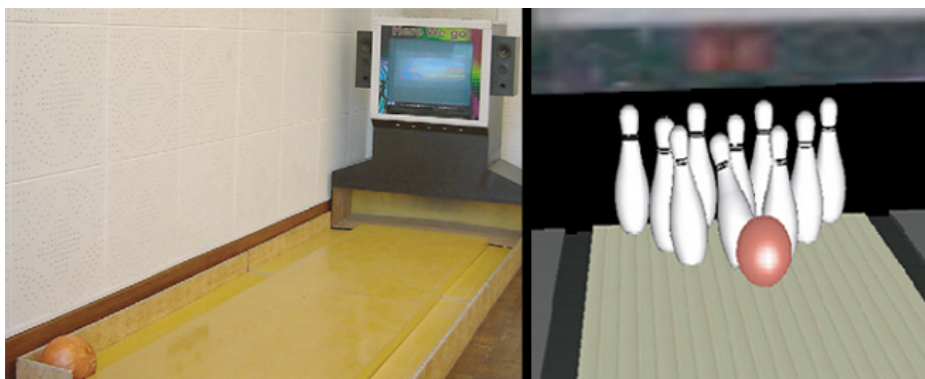


Figura 2.16: Exemplo do jogo Easybowling com a máquina (esquerda) e a simulação da colisão (direita) (Imagem retirada de [43])

Esta máquina permitia aos jogadores lançarem uma bola de bowling real numa pista de tamanho reduzido e o sistema detectava, através de câmaras, a velocidade e direcção dessa bola. Depois era gerada uma simulação e renderização em tempo real que mostrava o resultado dessa jogada e os pinos derrubados.

Para além destes exemplos temos também já vários exemplos de Exergames lançados no mundo dos videojogos. De seguida são apresentados alguns dos primeiros Exergames comerciais e também os que tiveram maior sucesso.

Podendo ser considerado o primeiro periférico para Exergaming existente, a Amiga Joyboard (ver Figura 2.17) foi um controlador lançado para a consola Atari 2600 no ano de 1982. Este acessório permitia substituir os botões direccionais do comando normal, bastando que o utilizador se inclinasse para o lado da direcção pretendida. O único jogo lançado para esta consola foi o jogo Mogul Maniac lançado exclusivamente para este acessório, e consistia num jogo de esqui[7].

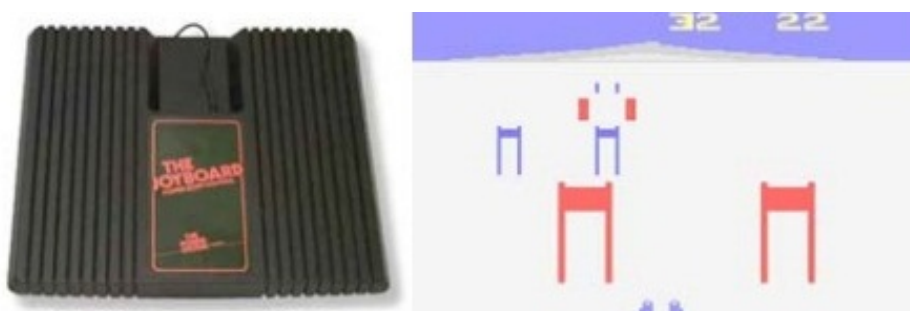


Figura 2.17: Joyboard (esquerda) e o jogo Mogul Maniac (direita)(Imagem retirada de [7])

No ano de 1987, foi lançado um acessório para a consola Atari 2600 chamado Foot

Craz (ver Figura 2.18), que consistia num controlador em forma de tapete e que pode ser considerado o predecessor do tapete do jogo Dance Dance Revolution. Foram apenas lançados dois jogos para este acessório que já chegou nos na fase final do ciclo de vida da Atari 2600, o que levou ao insucesso deste acessório e respectivos jogos[7].



Figura 2.18: Acessório Foot Craz (Imagem retirada de[7])

No final da década de 80 os videojogos estavam a entrar numa nova fase, muito por culpa do lançamento da popular consola Nintendo Entertainment System (NES) por parte da Nintendo. Foi lançado então um acessório para esta consola chamado Power Pad [65] (ver Figura 2.19), semelhante ao Foot Craz mas com maior complexidade. Foram lançados então vários jogos para este acessório em que era pedido ao jogador que pressionasse os botões numa certa ordem ou com uma certa cadência.



Figura 2.19: Nintendo Power Pad

Considerado como um dos primeiros grandes sucessos do Exergaming, o jogo Dance Dance Revolution ou DDR (ver Figura 2.20) [22] lançado em Abril de 1999 pela Konami

vendeu três milhões de cópias para a consola Playstation da Sony [23]. Este jogo apareceu nos salões de videojogos antes de seguir para os sistemas de entretenimento caseiros e a sua jogabilidade consiste em mover os pés segundo o ritmo que é pedido e pisando os botões correspondentes do tapete. Ainda continuam a ser produzidas novas versões deste videojogo para as consolas actuais.



Figura 2.20: DDR versão Arcade (esquerda) e o controlador para a versão Playstation (direita)

Em Julho de 2003 foi lançada a câmara EyeToy (predecessora da câmara PSEye) para a consola Playstation 2 com uma tecnologia que permitia jogar sem comandos num ambiente de realidade aumentada. O jogador via a sua imagem aparecer no ecrã juntamente com os objectos virtuais, e tinha que se movimentar de acordo com o pedido para completar as tarefas pretendidas. Saíram então vários jogos activos para serem usados com este acessório, que iam desde o simples movimento de limpar uma janela (como uma parte do jogo EyeToy:Play) a jogos dedicados exclusivamente a desporto, como o jogo EyeToy: Kinetic (ver Figura 2.21), criado em conjunto com a Nike MOTIONWORKS, e que oferecia uma vasta gama de disciplinas de fitness incluindo combate, aeróbica e ioga [57].

A consola Wii da Nintendo foi lançada em 2006 juntamente com o jogo Wii Sports e em 2007 foi lançado o jogo Wii Fit em separado, estes jogos podem ser considerados os Exergames com mais sucesso. O jogo Wii Sports foi mesmo o videojogo mais vendido de sempre, mas como foi lançado na mesma embalagem que a consola esse facto foi desvalorizado pelos criadores do jogo [35]. O jogo Wii Fit, lançado separadamente da consola Wii foi considerado o terceiro jogo mais vendido de sempre para as consolas com 22.61 milhões de cópias vendidas até Maio de 2010[36]. Este jogo foi lançado juntamente com a Wii Balance Board (ver Figura 2.22), acessório capaz de medir o

peso da jogador bem como o seu centro de gravidade.



Figura 2.21: Exemplo da jogabilidade do jogo EyeToy:Kinetic



Figura 2.22: Wii Balance Board

O jogo Wii Fit contém mais de 40 actividades físicas para exercitar o jogador, entre elas yoga, aeróbica, jogos de balanço e treino de força[10]. Em Outubro de 2009 foi lançada uma nova versão do jogo chamada Wii Fit Plus (ver Figura 2.23) [34] que contém mais exercícios que o primeiro jogo, um contador de calorias gastas e a possibilidade dos jogadores criarem um perfil que lhes fornece um regime de exercício físico adequado.

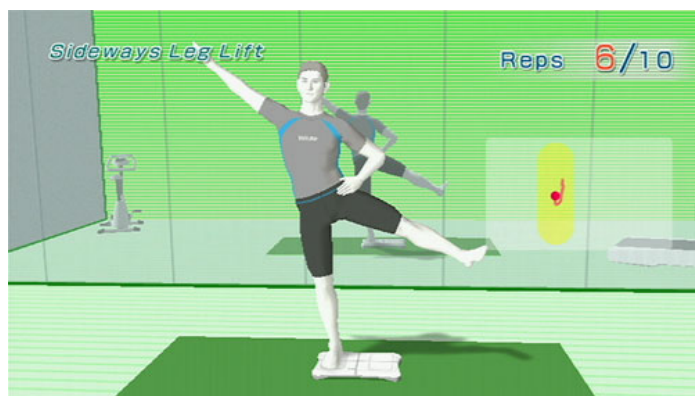


Figura 2.23: Ecrã de jogo Wii Fit Plus

Após ter lançado o jogo "EA Sports Active" para a consola Wii em 2009, a empresa de jogos EA Sports lançou uma segunda versão do jogo em Novembro de 2010, o "EA Sports Active 2" (ver Figura 2.24), para as consolas Wii, Playstation 3 e Xbox 360[14]. Este jogo usa as diferentes tecnologias de movimento de cada consola bem como um sensor para controlar o ritmo cardíaco. Estas tecnologias permitem a criação de uma experiência virtual mais completa a nível de exercício.

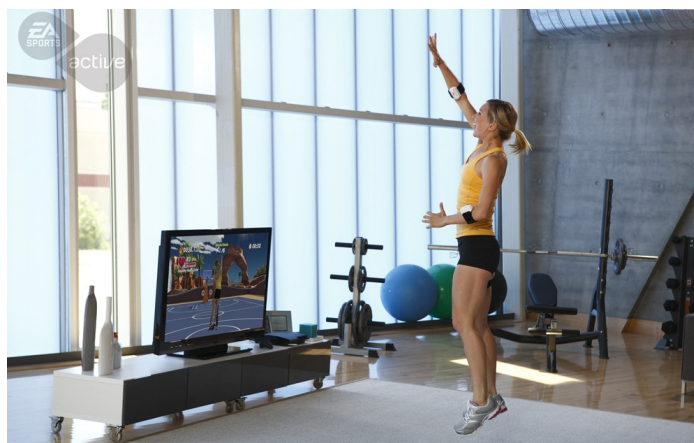


Figura 2.24: Exemplo do jogo EA Sports Active 2[14]

As grandes empresas de videojogos estão a apostar cada vez mais neste tipo de tecnologia, o que mostra que é um mercado com mais adesão por parte dos jogadores. Podemos afirmar, depois de observar estes exemplos, que os Exergames estão a ganhar cada vez mais importância na sociedade actual de videojogos e também na vida activa das pessoas.

2.4 OGRE: Exemplo de jogos

Apesar de não poder ser considerado um motor de jogo, o motor gráfico OGRE é usado na sua maioria para a criação deste tipo de entretenimento digital. Os jogos criados vão desde simples experiências a jogos comerciais e são de variados estilos de jogo. De seguida são apresentados alguns destes exemplos para mostrar as potencialidades do motor gráfico escolhido para a realização deste projecto.

Ankh [44]: É um jogo do género point-and-click e foi o primeiro jogo comercial a usar este motor gráfico, o qual foi desenvolvido pela empresa alemã Deck13 Interactive (ver Figura 2.25). Foi lançado em Novembro de 2005 e segue a história de Assil, um egípcio amaldiçoado por uma múmia na sua aventura para sobreviver. Foi um jogo bem recebido pela crítica tendo recebido 74 em 100 no site Metacritic [26] e já deu origem a uma sequência.



Figura 2.25: Screenshot do jogo Ankh

Earth Eternal [64]: Jogo do estilo MMORPG (Massive Multiplayer Online Role-Playing Game) que faz lembrar o famoso World of Warcraft. O Earth Eternal (ver Figura 2.26) foi desenvolvido pela empresa Sparkplay Media e é um jogo gratuito mas com bónus pagos. Neste momento o projecto encontra-se estagnado por falta de fundos e pertence à empresa de jogos Sankando.



Figura 2.26: Screenshot do jogo Earth eternal

Garshasp [13]: Jogo de acção na terceira pessoa (ver Figura 2.27) baseado num herói caçador de monstros da mitologia Iraniana, que foi lançado em Maio de 2011 e foi desenvolvido pela empresa de jogos Dead Mage Inc. Foi avaliado com a nota 49 em 100 pelo site Metacritic[27].



Figura 2.27: Screenshot do jogo Garshasp

Nimbus [39]: Jogo de corridas que teve o seu lançamento em 2010 pela empresa Noumenon Games (ver Figura 2.28). É um jogo que mistura corridas com resolução de puzzles e foi bem recebido pela crítica, tendo sido pontuado 80 em 100 pelo site Metacritic[29] e em 8 em 10 a nível gráfico pelo site IGN, que considerou o aspecto gráfico colorido e atractivo[41].



Figura 2.28: Screenshot do jogo Nimbus

Torchlight [47]: É provavelmente o jogo mais conhecido e com melhor recepção por parte dos sites de videojogos a usar o motor gráfico OGRE (ver Figura 2.29). É um jogo RPG de acção desenvolvido pela Runic Games e publicado pela empresa Perfect World. Na sua criação estiveram envolvidas as mentes criativas dos famosos jogos Diablo e Diablo 2. O jogo foi lançado em Outubro de 2009 para Windows e foi pontuado no site Metacritic com a fantástica pontuação de 83 em 100 [30].



Figura 2.29: Screenshot do jogo Torchlight

Através destes exemplos conseguimos perceber a versatilidade e qualidade do motor gráfico OGRE, motivos que ajudaram na escolha desta ferramenta para a criação do jogo mSquash.

Capítulo 3

Trabalho desenvolvido

3.1 Ferramentas usadas

Na criação de um videojogo podemos optar por diversas abordagens, podemos pensar em criar todo o jogo de raiz, opção mais complicada e que requer mesmo muito tempo, ou usar vários motores e bibliotecas existentes para ajudar na construção da aplicação. Essas ferramentas são extremamente importantes porque permitem a criação de um jogo mais evoluído num período de tempo mais curto, o ideal para este projecto. Foram por isso escolhidas as seguintes ferramentas de programação: o motor de renderização OGRE, as bibliotecas FMOD (som) e OpenCV (visão computacional). O jogo foi criado para correr sobre o sistema operativo Windows (nas versões Xp/Vista/Seven). A nível da interacção, o jogo foi criado para ser controlado com o comando Move da Playstation usando a câmara PSEye para fazer o seguimento do comando, permitindo assim uma experiência de simulação ao jogador. De seguida serão apresentadas estas ferramentas com mais detalhe.

3.1.1 OGRE

O OGRE (Object-Oriented Graphics Rendering Engine)[62] é, como o nome indica, um motor de renderização gráfica 3D open source (LGPL) e multiplataforma. Foi desenvolvido na linguagem C++ e é um motor organizado pelos princípios da programação orientada a objectos. Para além de ser um motor muito bem organizado a nível de código e com grandes potencialidades, o OGRE está constantemente a ser alvo de actualizações e de correcções de erros por parte da equipa de desenvolvimento.

Outro factor importante é o facto de possuir uma vasta comunidade de utilizadores que estão constantemente a desenvolver novas ferramentas para complementar este motor, e consequentemente ajudar o resto da comunidade.

O OGRE não pode ser considerado um motor de jogo, mas sim uma ferramenta que fornece uma solução para representar o jogo graficamente, podendo até ser usado para outras aplicações fora do mundo dos videojogos. Para criar um videojogo temos por isso que acrescentar diversas bibliotecas ao OGRE, como bibliotecas de som, inteligência artificial, física, entre outras.

Como se baseia em objectos, o objecto Root é considerado o objecto de entrada e organizador do sistema OGRE, é o primeiro a ser inicializado, e é onde são criados os objectos de alto nível tais como os gestores de cenas (Scene Managers), os plugins, os sistemas de render e a janela de renderização, entre outros. Neste objecto é definida a linguagem gráfica a ser usada (DirectX ou OpenGL) e são criadas as conexões com os periféricos gráficos (monitor e placa gráfica). A maioria das classes do OGRE pode ser enquadrada num de 3 tipos[59]:

- **Gestão de cenas (Scene Management):** Estas classes são usadas para gerir a cena de renderização e a sua estrutura, o posicionamento e orientação das câmaras, as luzes, a posição dos objectos, etc.
- **Gestão de recursos (Resource Management):** Os recursos necessários para gerar uma renderização gráfica, tais como os tipos de letra, as texturas, a geometria da cena (as malhas dos objectos virtuais) são geridos pelas classes deste tipo, que os carregam, manipulam e eliminam.
- **Renderização (Rendering):** As classes deste tipo são responsáveis por mostrar a imagem final, trabalham por isso com o nível inferior da renderização. Estas classes são usadas pelas classes do tipo Gestão de Cenas para estas mostrarem os seus conteúdos no ecrã.

Depois de criada a OGRE Root criamos através dela um gestor de recursos onde são carregados todos os recursos necessários para a cena. É depois criado um gestor de cenas (SceneManager) para gerir os objectos virtuais do jogo. São adicionados os nós (nodes) dos objectos que queremos gerir na cena, tal como as malhas e as luzes. Depois de estar tudo posicionado como queremos na cena renderizamos uma frame e

obtemos a nossa cena em 3D. No jogo basta actualizar as posições e orientações dos nossos nós e renderizar uma frame a cada ciclo de jogo.

Na Figura 3.1 temos um esquema dos objectos principais do OGRE e como se relacionam.

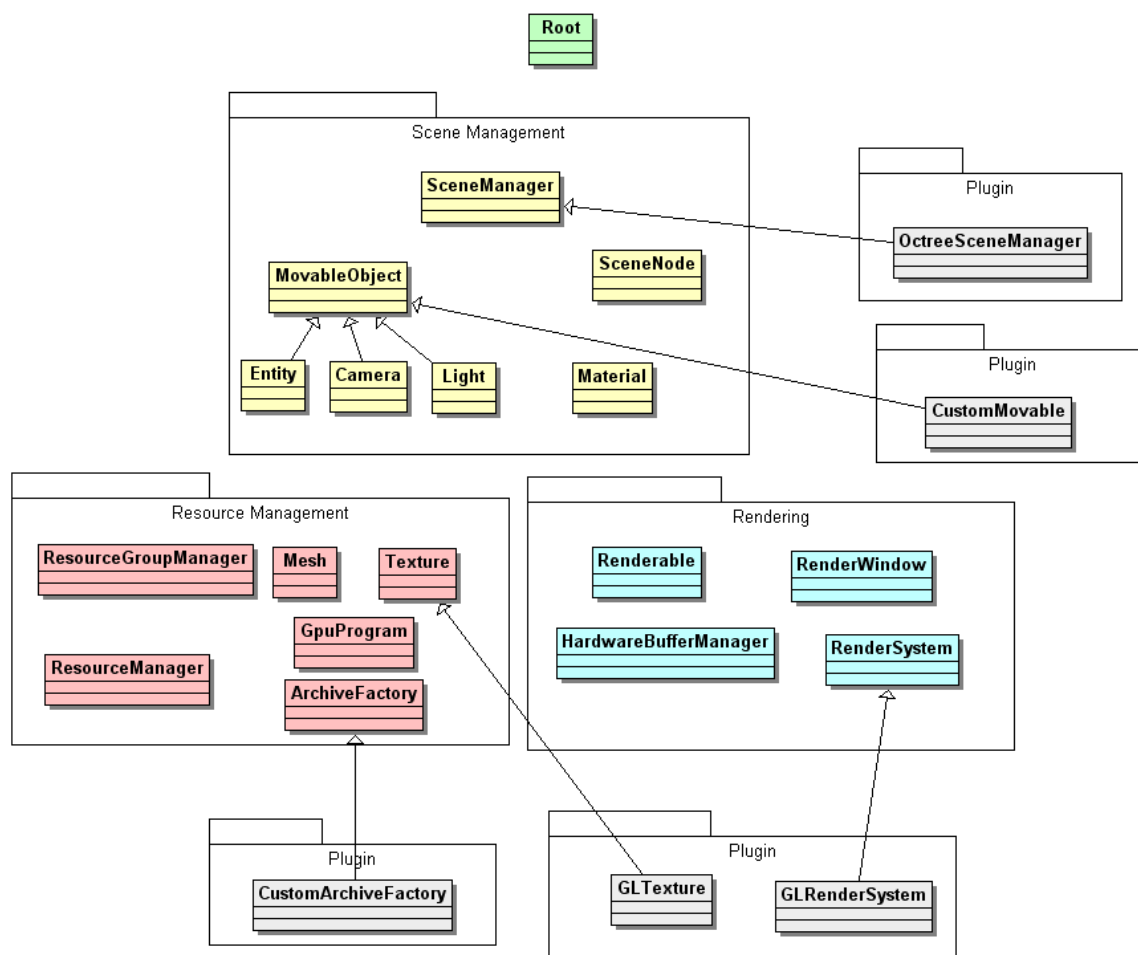


Figura 3.1: Principais objectos do OGRE (retirado de [59])

3.1.2 FMOD

FMOD [18] é uma biblioteca de áudio usada em diversas aplicações e que permite que sejam reproduzidos ficheiros de sons em diversos formatos nessas aplicações, entre outras características avançadas de áudio. Os produtos da FMOD são gratuitos se forem usados em produtos não comerciais e dividem-se essencialmente em 4 tipos: FMOD Ex Programmers API, FMOD Designer, FMOD Sandbox e FMOD UE3 Integration. No nosso projecto foi usado o FMOD Ex Programmers API, que permitiu a reprodução de

sons no jogo, como o barulho das colisões da bola, o som de fundo e as manifestações do público relacionadas com a performance do jogador.

Para instalar o FMOD no jogo mSquash foi preciso fazer download do instalador do FMOD Ex API[60] que após a sua instalação nos fornece diversas bibliotecas para serem usadas. De seguida é apresentado um exemplo que mostra como reproduzir um som numa aplicação baseado no uso deste motor no jogo mSquash.

Para usar este motor de áudio primeiro foram incluídos os cabeçalhos "fmod.h" e "fmod_errors.h" na nossa aplicação. Foram depois criadas as variáveis para o sistema de som, para o som a executar e para o canal pretendido.

```
FMOD_SYSTEM *sound_system; // Criar um sistema de som
FMOD_SOUND *sound1; //Som para reproduzir
FMOD_CHANNEL *channel = 0; //Canal para reproduzir o som
```

De seguida é criado e iniciado o sistema de som.

```
FMOD_System_Create(&sound_system); //Cria o sistema de som
FMOD_System_Init(sound_system, 32, FMOD_INIT_NORMAL, NULL); //Inicia o sistema de som
```

Precisamos depois de carregar o som pretendido ("sound.mp3") e definir o seu modo.

```
FMOD_System_CreateSound(sound_system, "sound.mp3", FMOD_HARDWARE, 0, &sound1);
// Carregar o som em format mp3
FMOD_Sound_SetMode(sound1, FMOD_LOOP_OFF); //Desligar o modo Loop
```

Antes de poder reproduzir qualquer som devemos actualizar o motor de áudio a cada ciclo de jogo, por isso colocamos no método do ciclo de jogo a seguinte linha de código.

```
FMOD_System_Update(sound_system);
```

Por fim podemos reproduzir o nosso som bastando para isso chamar a seguinte linha de código.

```
FMOD_System_PlaySound(sound_system, FMOD_CHANNEL_FREE, sound1, 0, &channel); //Reproduzir o som sound1
```

3.1.3 OpenCV

OpenCV (Open Source Computer Vision)[42] é uma biblioteca usada para o desenvolvimento de aplicações na área da computação visual. Foi criada pela Inter Corporation no ano 2000 e possui licença BSD. Foi originalmente escrita na linguagem C mas já possui uma interface em C++. Esta biblioteca pode ser usada em várias áreas da computação visual, como por exemplo na área da interacção Homem-Computador (HCI); no reconhecimento de gestos; identificação de objectos; segmentação entre outras.

No caso do jogo mSquash o OpenCV foi usado para mostrar os ecrãs de arranque do jogo e para processar em tempo real as imagens resultantes da câmara PSEye, trabalhando-as de modo a poder ser feito o mapeamento da bola do comando Move.

Após instalar o executável do site oficial do openCV para Windows e configurar a nossa aplicação podemos usar as suas bibliotecas adicionando os seguintes cabeçalhos ao código: "cv.h" e "highgui.h". De seguida é criada uma captura de câmara para obter as frames da câmara Playstation Eye e uma imagem (frame) que é criada automaticamente com o tamanho da imagem que é capturada.

```
CvCapture* capture = cvCreateCameraCapture(0);  
IplImage* frame = cvQueryFrame( capture );
```

Para substituir a imagem que está na variável "frame" podemos a qualquer momento chamar a seguinte função que captura a imagem actual da câmara.

```
frame = cvQueryFrame( capture );
```

No capítulo 3.5 da dissertação é explicado como é feito o mapeamento da bola usando os filtros e funções da biblioteca OpenCV.

3.1.4 Periféricos

Para o jogo mSquash foram usados dois periféricos da marca Sony Playstation: o comando de movimento Playstation Move[58] e a câmara Playstation Eye[55]. Estes dois periféricos combinados permitem que sejam obtidos dados no jogo que podem ser usados para posicionar e orientar a raquete virtual, permitindo assim uma experiência de simulação do desporto squash.

Fabricante	SCEE
Número de modelo	SLEH-00203
Data de lançamento	Dezembro de 2007
Cor	Preto
Interface	USB 2.0
Dimensões (aprox.)	84 x 67 x 57 mm
Peso (aprox.)	173g
Gravação de vídeo	640x480 @60 f/s
Campo de visão	Zoom de 56° a 75° FOV

Tabela 3.1: Especificações da câmera PSEye[55]

A câmera Playstation Eye (ver Figura 3.2 e Tabela 3.1), lançada em 2007 para a consola Playstation 3, é a sucessora do periférico EyeToy da consola Playstation 2. Criada para introduzir a visão computacional (CV) como maneira de interagir com os jogos da Playstation 3, tais como no campo da realidade aumentada, detecção de cores e de limites e mapeamento facial. Possui também um microfone capaz de ser usado para localização de voz, supressão de ruído de fundo, cancelamento de eco e reconhecimento de voz. Para complementar esta tecnologia foi lançado em Setembro de 2010 o comando Move, que veio melhorar a experiência de interação dos jogadores com os jogos da Playstation 3.



Figura 3.2: Playstation Eye

Fabricante	SCEI
Número de modelo	CECH-ZCM1
Data de lançamento	Setembro de 2010
Cor	Preto
Interface	USB 2.0 e Bluetooth
Dimensões (aprox.)	200 mm (altura) x 46 mm (diâmetro)
Peso (aprox.)	145g
Número de botões	9
Alimentação	Bateria de Li-ion
Modo de carga	Por USB

Tabela 3.2: Especificações do comando Playstation Move[58]

O comando Playstation Move (ver Figura 3.3 e Tabela 3.2) é um controlador de jogos com sensores de movimento para a Playstation 3. Este comando contém uma bola de plástico que emite luzes de várias cores. Foi lançado em Setembro de 2010 e funciona em conjunto com a câmara Playstation Eye que faz o mapeamento do comando através da bola iluminada e, permite assim calcular o diâmetro da bola e saber a sua posição no espaço em que se encontra, fornecendo assim dados importantes para o jogo.



Figura 3.3: Playstation Move

O comando Move possui também vários sensores que ajudam na virtualização dos movimentos executados com o comando pelo jogador. Os acelerómetros presentes neste comando captam as forças que são aplicadas ao comando segundo os 3 eixos do espaço de tridimensional e transformam-nas em valores. Os sensores de rotação captam as rotações aplicadas ao comando também nos 3 eixos e devolvem as alterações angulares nos 3 eixos. Por fim temos os magnetómetros, que funcionam medindo a intensidade e direcção do campo magnético terrestre, ajudando assim na orientação do objecto virtual controlado pelo comando.

Além disso o comando Move também conta os botões mais conhecidos dos comandos DualShock3 da Playstation (Triângulo, Quadrado, Circulo e X, Select ,Start e botão PS) e com dois botões criados exclusivamente para este comando, o botão Move e o botão Trigger. Deste modo os criadores de jogos podem adaptar os seus jogos já lançados ao comando Move com facilidade.

O comando Move foi recebido com críticas positivas por parte dos sites especialistas em videojogos, tendo sido analisado com boas pontuações[28]. Foi considerado por vários sites como uma evolução tecnológica em relação ao comando Wii da Nintendo por ser mais preciso e realista.



Figura 3.4: Screenshot do jogo Under Siege

A 6 de Junho de 2011 foi anunciado por parte da Sony que o comando Playstation Move já vendeu 8.8 milhões de unidades por todo o mundo[16]. Na altura do seu lançamento, o comando Move foi criticado por os seus jogos compatíveis não se equipararem à qualidade do comando em si. Mas com o lançamento de novos títulos compatíveis com o comando move (como o jogo Killzone 3), juntamente com a adaptação de jogos

já existentes a esta tecnologia (Heavy Rain) as opiniões sobre a qualidade dos jogos jogáveis com o Move foi-se alterando.

Um bom exemplo é o jogo português Under Siege (ver Figura 3.4), desenvolvido pela empresa portuguesa Seed Studios[49], que foi disponibilizado para download desde o dia 02 de Junho de 2011 e já é compatível com o comando Move, tendo recebido várias críticas positivas por parte da imprensa de videojogos[31].

Após o aparecimento do comando Move, foram surgindo vários acessórios (ver Figura 3.5) que permitem tornar a experiência de jogo mais imersiva, dando ao jogador a sensação de estar a segurar numa espada por exemplo, a jogar ping-pong com uma raquete real ou mesmo podendo segurar uma replica de arma em tamanho real, como o novo acessório oficial Sony Playstation denominado Playstation Move Sharpshooter[56].



Figura 3.5: Vários acessórios para o comando Playstation Move

3.2 O jogo mSquash

Foi criado um jogo baseado no squash usando o comando Move da Playstation para simular o comportamento da raquete. Esta decisão deveu-se ao facto de existirem vários desportos de raquete com comportamento algo semelhante, o que permite que haja uma maior facilidade na adaptação ao jogo por parte dos jogadores. Outra razão para a escolha deste jogo foi que seria mais fácil criar uma boa simulação que envolvesse o

jogador de um modo mais eficaz usando o desporto squash do que o ténis por exemplo, já que teria que ter um jogador controlado pelo computador com um comportamento realista e, isso seria uma tarefa mais complicada e morosa, logo optou-se por uma alternativa mais realista de acordo com as necessidades e o tempo disponível.

Outro dos motivos para a escolha do squash foi o facto de ser um desporto bastante activo (ver Figura 3.6), e assim também poderíamos abordar o tema dos jogos que envolvem exercício físico, também designados de Exergames, actualmente muito na moda. Note-se que o squash é um desporto que já foi considerado o desporto mais saudável no artigo de Neal Santelmann para a Forbes Magazine [48]. Foi por isso dado o nome mSquash ao jogo, nome que vem da conjugação da tecnologia usada e do desporto que se pretendia simular (move + Squash).



Figura 3.6: Jogo de squash (Imagem retirada de [4])

3.2.1 Regras de jogo

Para criar uma experiência de jogo mais variada foram criados três modos de jogo, um de treino em que o jogador não é avaliado, um modo em que o jogador é avaliado de acordo com a zona da parede de fundo em que acerta com a bola (modo Normal) e por último o modo de jogo Target que consiste num jogo de squash em que existe um alvo semelhante ao jogo de Dardos e em que o jogador só pontua caso acerte no alvo com a bola.

O modo de treino (ver Figura 3.7) foi criado para permitir aos jogadores uma primeira

experiência com o jogo e com a interação utilizada no jogo recorrendo ao comando Move, de modo a que possam treinar um pouco antes de avançar para os seguintes modos de jogo. Este modo tem a duração de 5 minutos, duração que foi escolhida por ser considerada suficiente para que os jogadores percebessem os objectivos do jogo e se adaptassem minimamente à sua jogabilidade. Neste modo de jogo os jogadores podem jogar o número de bolas que conseguirem, terminando o jogo assim que os 300 segundos (5 minutos) se esgotem .

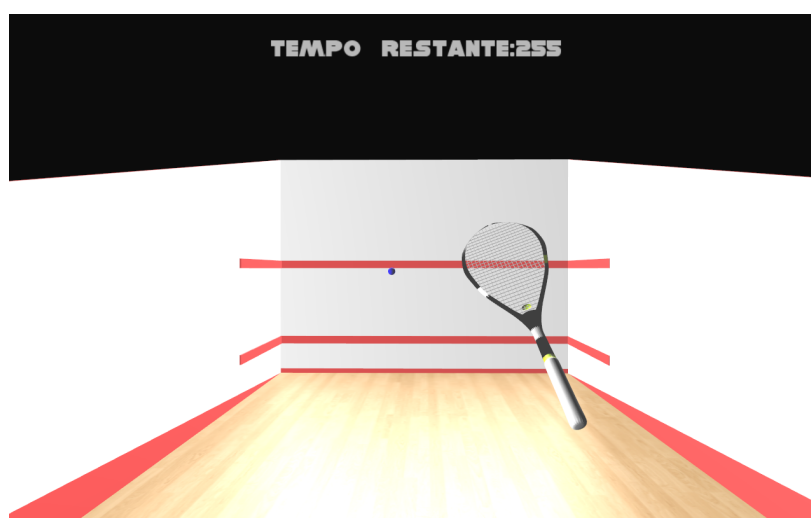


Figura 3.7: Modo de jogo Treino

O modo Normal foi criado para avaliar a prestação dos jogadores de acordo com as zonas em que a bola acerta na parede de fundo, sendo estas zonas limitadas pelas 2 linhas vermelhas da parede como mostra a Figura 3.8.

O jogador preenche um questionário antes de o jogo arrancar onde define para além de outra informação o número de bolas a jogar, após esgotadas essas bolas o jogo mostra a pontuação final do jogador, bem como a pontuação máxima por bola e a média por bola.

Na Figura 3.8 temos um exemplo do jogo no modo normal onde se pode ver que o jogador tem 3 pontos na bola actual, um total de 95 pontos nas bolas jogadas, está no nível de dificuldade 2 e na bola numero 8 em 15 possíveis.

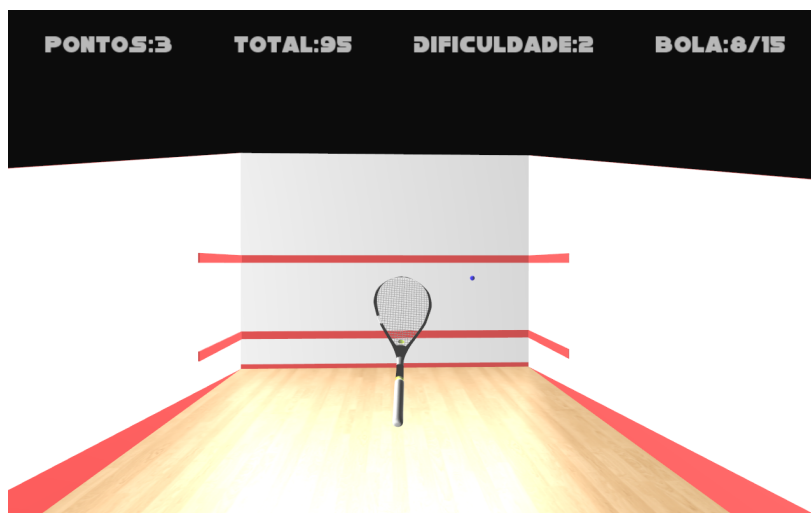


Figura 3.8: Modo de jogo Normal

Após ter sido criado o modo Normal achou-se que seria útil criar um modo de jogo mais difícil e desafiante em que fosse precisa maior precisão para obter pontos, o modo Target. Do mesmo modo, o jogador preenche o questionário inicial e escolhe o número de bolas que terá à sua disposição para jogar e para usar de modo a acertar no alvo. O alvo tem 12 posições aleatórias possíveis, que mudam sempre que a bola atinge o alvo.

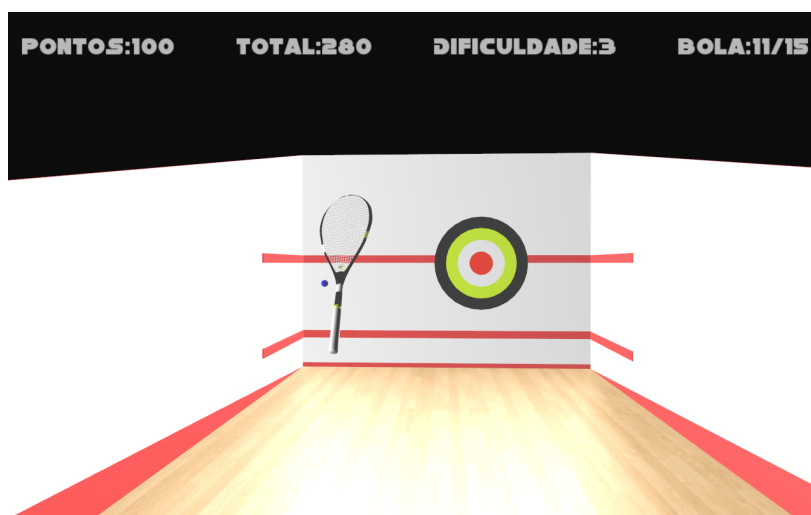


Figura 3.9: Modo de jogo Target

A Figura 3.9 mostra um exemplo do modo de jogo Target em execução, na qual se vê o alvo na parede de fundo e as três zonas pontuáveis: vermelho, branco e verde

alface. Podemos também observar que a informação de jogo nos mostra que o jogador tem 100 pontos na bola actual, um total de 280 pontos e que está a jogar a bola 11 em 15 no grau de dificuldade 3.

3.2.1.1 Sistema de pontuação

Para pontuar e avaliar a performance do jogador foi criado um sistema de pontuação para cada tipo de jogo e foi elaborado um relatório final de sessão contendo várias informações. Para o jogo no modo treino o jogador não é pontuado e não é criado relatório. Isto porque o modo de treino serve apenas para o jogador se habituar ao jogo e ao modo de interacção para que se possa sentir mais à vontade nos outros modos de jogo.

No modo normal o jogo é pontuado da seguinte forma: 1 ponto se a bola acerta na zona acima da linha vermelha superior da parede ou se acerta na zona abaixo da linha inferior e 2 pontos se o jogador acerta com a bola na zona intermédia entre as duas linhas vermelhas (como mostra a Figura 3.10).

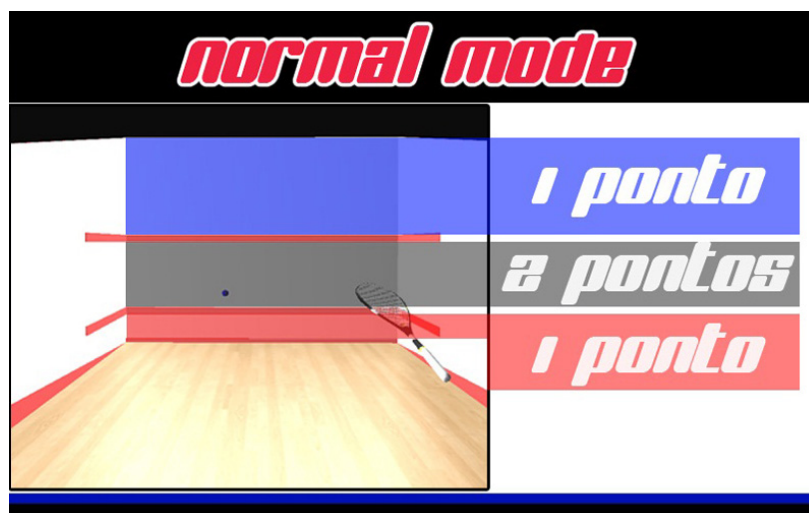


Figura 3.10: Instruções da pontuação no modo Normal

A pontuação do modo Target (ver Figura 3.11) depende da zona do alvo em que a bola acerta, se acertar no centro do alvo (zona vermelha) o jogador ganha 50 pontos. Caso acerte na zona branca ganha 25 pontos e apenas 10 na zona amarela.

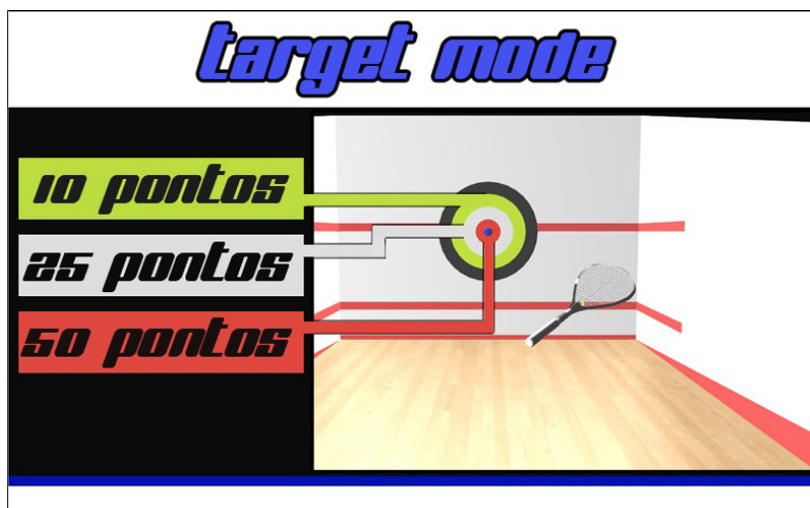


Figura 3.11: Instruções da pontuação no modo Target

De modo a que se possa estudar a precisão do jogador com o comando foram criados 3 níveis de dificuldade (ver Figura 3.12): Fácil (1), Médio (2) e Difícil (3). Cada um desses níveis tem uma área diferente para a raquete, o que indica que é preciso muito menos precisão com o comando para acertar na bola no nível fácil do que no nível médio, sendo difícil o nível que necessita de maior precisão.

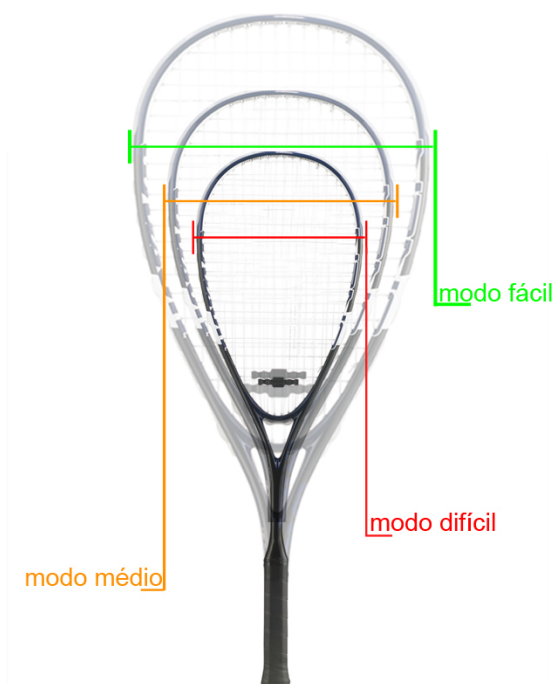


Figura 3.12: Exemplo dos vários níveis de dificuldade no jogo

3.2.1.2 Relatório de sessão

Para poder avaliar cada sessão de jogo foi criado um relatório de texto com os valores pessoais do jogador e com vários parâmetros que podem indicar ou não padrões de comportamento relativamente ao jogo. O relatório será explicado em mais detalhe no capítulo 4 da dissertação, onde será feita a interpretação dos resultados provenientes dos vários testes realizados com jogadores para o qual foram gerados os respectivos relatórios.

3.3 mSquash: Estrutura do código

Um dos primeiros passos na criação de um videojogo é a escolha da linguagem de programação a usar. A linguagem C++ tem sido muito usada no mundo dos videojogos, por ser uma linguagem orientada a objectos, rápida e com muitas potencialidades. A escolha desta linguagem também se deve ao facto de ser a linguagem base do motor de render Ogre, o motor base de toda a aplicação. O código do jogo foi criado usando várias classes, objectos, bibliotecas e cabeçalhos que interagem entre si, como mostra o esquema da Figura 3.13.

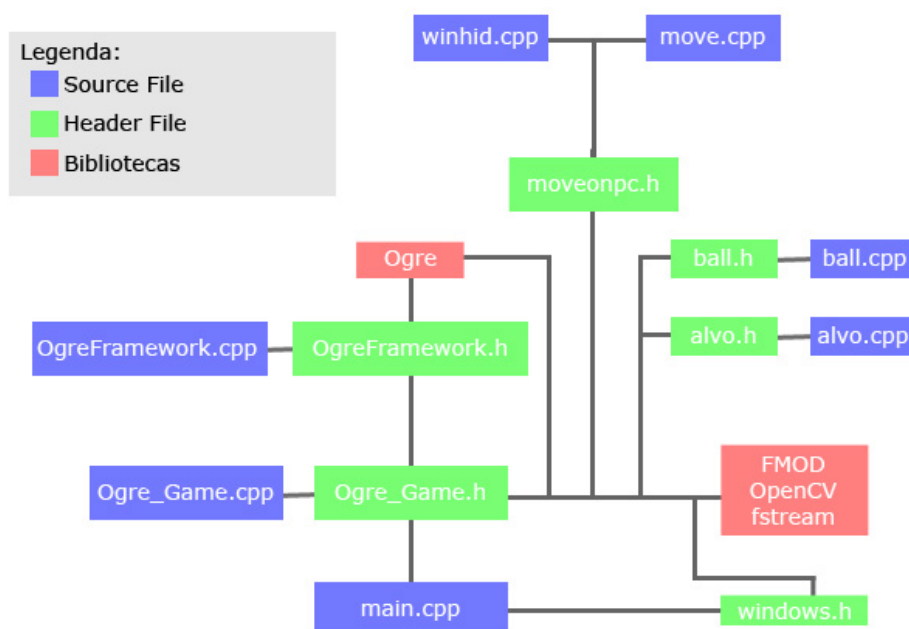


Figura 3.13: Esquema do código do jogo mSquash

A classe main.cpp é a classe principal de execução do código, incluindo o cabeçalho

da classe `Ogre_Game` (`Ogre_Game.hpp`) para que possa criar um objecto desse tipo e executar o código de jogo. O cabeçalho da classe `Ogre_Game` inclui as bibliotecas de som, captura de imagem, tempo e também as bibliotecas do motor de render `Ogre3D`. Esta classe usa também uma prévia implementação chamada `OgreFramework` que consiste num código base preparado para criar um motor de jogo usando o `Ogre3D`[40]. Também é incluído na classe `Ogre_Game` o cabeçalho `moveonpc.h`, que juntamente com as classes `winhid.cpp` e `move.cpp` contém as funções para capturar e enviar dados de e para o comando `Move`. As classes `ball` e `alvo` são classes que servem para criar e manipular objectos do tipo bola e alvo. As classes são explicadas com mais detalhe no decorrer deste capítulo.

Bibliotecas são ficheiros que, após adicionados, permitem que sejam usadas funções previamente criadas sem que seja preciso alterar ou até conhecer o código das mesmas. Neste jogo foram usadas várias bibliotecas para vários fins, sendo apresentadas de seguida as mais importantes:

- **OGRE:** Estas bibliotecas permitem usar as funções do motor de renderização `Ogre 3D`, tal como a biblioteca "`OgreOverlay.h`" que permite a criação de uma `overlay` e que serviu para mostrar as informações ao jogador.
- **FMOD:** Bibliotecas de som que permitem adicionar som a qualquer aplicação, permitindo assim que sejam usadas funções para carregar e reproduzir sons no jogo.
- **OpenCV:** As funções desta biblioteca permitem capturar informação da câmara de vídeo para ser usada no mapeamento e também para criar e mostrar os ecrãs de configuração inicial.
- **Fstream:** Biblioteca que fornece uma interface para ler e escrever dados para ficheiro, permitindo assim a criação do nosso relatório de jogo e também que fossem guardados e carregados os dados de sessão.
- **Time.h:** Permite que sejam usadas funções que manipulam o tempo, usada neste caso para cronometrar o jogo e para limitar e calcular o número de frames por segundo.

De seguida são explicadas algumas das principais classes.

- `Main.cpp`: A classe `main.cpp` é a classe do código que é executada no início do jogo. Esta classe inicia uma aplicação do Ogre, cria uma instância da classe `Ogre_Game` e executa a função `startGame()` para iniciar o jogo. Quando a aplicação do Ogre é terminada esta classe desliga os comandos Move e termina.
- `OgreFramework.cpp`: A classe `OgreFramework` é a base de um projecto OGRE que é baseada num tutorial do site oficial do OGRE [40]. Nesta classe são criados vários objectos essenciais para o funcionamento do OGRE como a câmara e as suas características, a luz ambiente, a criação de dispositivos de entrada e de uma janela de render.
- `Winhid.cpp` e `move.cpp`: Estas duas classes foram baseadas em classes retiradas do projecto "moveonpc"[33] e permitem que se consiga interagir com o comando Move, tanto recebendo os dados do mapeamento dos botões e dos sensores de movimento como enviando informação ao mesmo para alterar a luz e a vibração, por exemplo.
- `Ball.cpp` e `alvo.cpp`: Estas classes representam objectos que são criados no jogo e os seus métodos são funções que os afectam. A classe `ball.cpp` corresponde à bola de jogo e a classe `alvo.cpp` corresponde ao alvo do tipo de jogo Target Mode.
- `Ogre_game.cpp`: Pode ser considerada a classe mais importante do código, porque é nela que estão contidos os métodos mais importantes do jogo, tal como a configuração da aplicação, o ciclo de jogo e a lógica do mesmo. A classe começa por criar uma ficha de utilizador, por configurar a câmara e o comando Move e por carregar os objectos a serem renderizados (bola, cenário, luzes, etc.). Arranca depois o ciclo de jogo, onde são chamadas as funções de tracking e leitura dos sensores do comando, de actualização da física de jogo, e caso tenha terminado o jogo, faz o relatório final e mostra os dados finais de sessão no ecrã.

3.4 Motor de física

O motor de física do jogo `mSquash` foi criado de raiz, com o intuito de simular o movimento de uma bola num espaço restrito e das colisões da mesma com a raquete. Para isso foram usadas diversas variáveis para simular um comportamento semelhante

ao real. Foram definidos 3 objectos principais, o campo de Squash, a raquete e a bola. Para a bola temos 3 variáveis de posição (ver Figura 3.14), uma para cada eixo:

- Pos_bola_x: Posição no eixo X
- Pos_bola_y: Posição no eixo Y
- Pos_bola_z: Posição no eixo Z

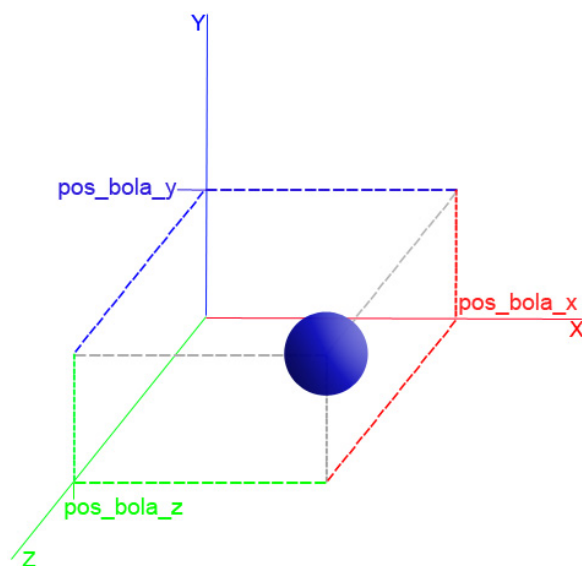


Figura 3.14: Representação da posição da bola

Estas posições vão ser actualizadas a cada frame do motor de render, neste caso sendo actualizadas 30 vezes a cada segundo (30 fps). Para alterar estas posições temos definidas variáveis de velocidade da bola para cada eixo (ver Figura 3.15):

- Vel_bola_x: Componente X da velocidade
- Vel_bola_y: Componente Y da velocidade
- Vel_bola_z: Componente Z da velocidade

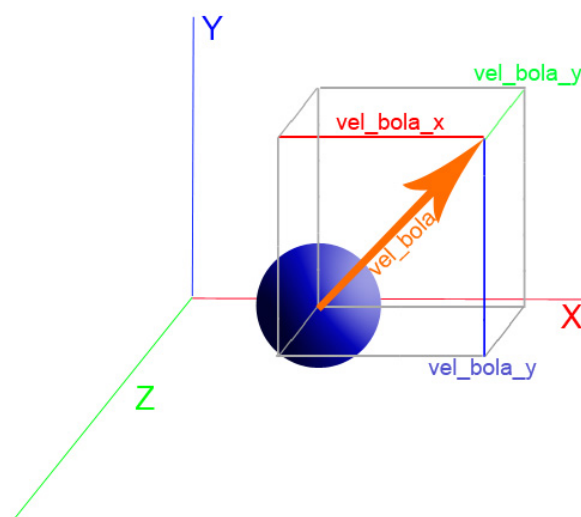


Figura 3.15: Representação das velocidades da bola

Para as velocidades não serem constantes, o que resultaria num movimento infinito da bola, foram criadas várias forças para simular o comportamento de uma bola num sistema físico real:

- `for_grav`: Força gravítica aplicada à bola segundo o eixo Y
- `atr_ar`: Força do atrito do ar aplicada a todos os eixos da velocidade da bola
- `atr_col`: Força para simular a perda de velocidade que ocorre numa colisão

A força gravítica, por exemplo, será apenas aplicada à componente Y da velocidade bola. A cada ciclo de jogo essa componente vai sofrer uma aceleração negativa em relação a Y (considerando o eixo positivo no sentido de baixo para cima).

- $vel_bola_y(nova) = vel_bola_y(antiga) - for_grav$

A força de atrito é aplicada às 3 componentes da velocidade para simular o atrito do ar, sendo aplicada uma aceleração contrária ao movimento da bola. Para o caso da componente X temos, por exemplo:

- Se $vel_bola_x > 0$ temos $vel_bola_x(nova) = vel_bola_x(antiga) - atr_ar$
- Se $vel_bola_x < 0$ temos $vel_bola_x(nova) = vel_bola_x(antiga) + atr_ar$

A raquete, sendo a representação virtual do nosso comando, tem que ter um efeito similar à realidade física aquando da colisão com a bola. Foram definidas então as seguintes variáveis para a raquete:

- Pos_raq_x: Posição da raquete em X
- Pos_raq_y: Posição da raquete em Y
- Pos_raq_z: Posição da raquete em Z

Estes valores são obtidos através do tracking feito com a câmara à luz da bola do comando Move, e são actualizados a cada 1/30 segundos. Os valores da pos_raq_x e pos_raq_y são obtidos de acordo com a posição da bola do comando move. O valor pos_raq_z é calculado com base no diâmetro da bola do comando Move.

Como temos a posição da raquete em cada frame podemos facilmente calcular a sua velocidade segundo cada um dos eixos, bastando para isso subtrair à posição da frame actual a posição da frame anterior, assim obtendo:

- $Vel_raq_x = Pos_raq_x - pos_raq_x_old$: Componente X da velocidade
- $Vel_raq_y = Pos_raq_y - pos_raq_y_old$: Componente Y da velocidade

Usando estas velocidades podemos alterar o movimento da bola de um modo mais realista quando a raquete colidir com ela. Não foi calculada a componente Z da velocidade da raquete porque se resolveu atribuir à bola a força correspondente ao valor do acelerómetro, dando assim uma maior realidade ao jogador aquando da pancada.

Na colisão entre a bola e o cenário (ver Figura 3.16) é usada a função troca_dir_velocidade() com os parâmetros eixo e com o valor do atrito de colisão. Se a bola colidir com a parede do fundo, por exemplo, é chama a função com o eixo Z e com o valor definido como atrito de colisão. Esta função começa por trocar o sentido da velocidade:

- $Vel_bola_z = - vel_bola_z$

E depois tira-lhe o valor correspondente ao atrito de colisão:

- $Vel_bola_z = vel_bola_z - atr_col$

O atrito de colisão permite assim simular uma colisão parcialmente elástica. Se for uma colisão com uma parede lateral é chamada a função com o parâmetro X e se for colisão com o chão é chamada com o parâmetro Y.

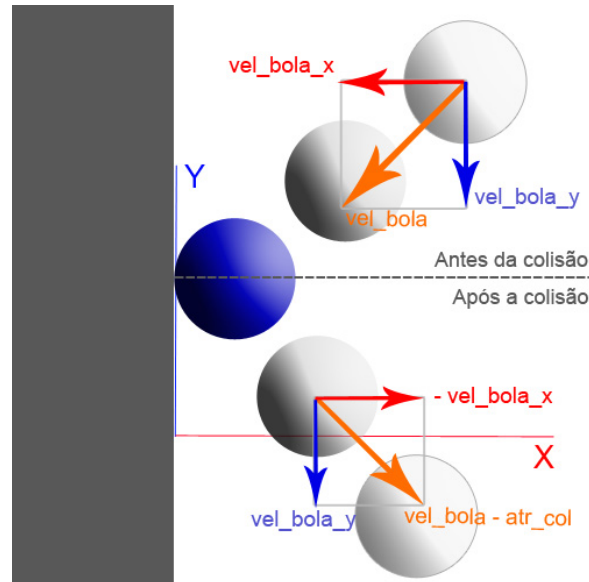


Figura 3.16: Esquema da colisão da bola com a parede

Quando é verificado que a bola se encontra na mesma zona no espaço do que a raquete é chamada a função `colisão_raquete()` da classe `bola` com os parâmetros: a velocidade da raquete para os eixos X e Y e a força do acelerómetro do comando `move` para Z. Esta função é chamada no código do seguinte modo:

```
if (fabs (m.RawForceZ) > fabs (m.RawForceX) ) {
    bola.colisao_raquete (vel_raq_x, vel_raq_y,
        -fabs (m.RawForceZ/forca_gen) );
} else {
    bola.colisao_raquete (vel_raq_x, vel_raq_y,
        -fabs (m.RawForceX/forca_gen) );
}
```

Em que `vel_raq_x` e `vel_raq_y` são as componentes X e Y da velocidade da raquete e `m.RawForceZ` e `m.RawForceX` são as forças dadas pelos acelerómetros segundo Z e X respectivamente. A condição inicial (`if`) serve para que seja dada à bola a maior

força entre as forças X e Z do comando, para que o jogador possa "bater" na bola com a rotação que lhe dá mais força aplicada sendo escolhida a maior entre elas. A variável `forca_gen` é uma variável que depende do género do jogador e que ajusta a força de acordo com esse parâmetro. Foi introduzida porque os jogadores masculinos tinham tendência para aplicar mais força e as jogadoras menos, assim pretende-se encontrar um equilíbrio entre todos os jogadores. A função `fabs()` serve para encontrar o valor absoluto do parâmetro, visto que o valor da força tanto pode ser negativo como positivo de acordo com o lado do comando que se usa para bater na bola.

3.5 Tracking

Usando o comando `Move` e uma câmara permite-nos construir uma função que faz o tracking da posição da bola do comando no espaço, de modo a que esses dados possam ser usados para movimentar os nossos objectos virtuais. Para isso foi usada a biblioteca multi-plataforma `OpenCv` (Open Source Computer Vision Library), que permite aceder às frames da câmara, filtrar essas frames e executar o algoritmo de reconhecimento.



Figura 3.17: Captura original da câmara

É criada uma captura usando a biblioteca `openCv` que permite fazer "query" para obter a frame actual, neste caso é usado o comando `cvCreateCameraCapture(0)`; para usar a câmara por defeito do sistema. Depois em cada ciclo de jogo (frame) vai ser captada a imagem que a câmara está a captar no momento (ver Figura 3.17).

```
frame = cvQueryFrame( capture );
```

São aplicados depois vários filtros que permitem que a imagem filtrada seja uma imagem mais fácil de ser usada para tracking. É executado um filtro `cvSmooth` que suaviza a imagem com um efeito "Blur"(ver Figura 3.18):

```
cvSmooth(frame, frame, CV_BLUR, 3 , 0, 0, 0);
```



Figura 3.18: Imagem suavizada

De seguida é aplicado um filtro de Threshold (ver Figura 3.19):

```
cvThreshold( frame, frame, 230, 240, CV_THRESH_BINARY);
```

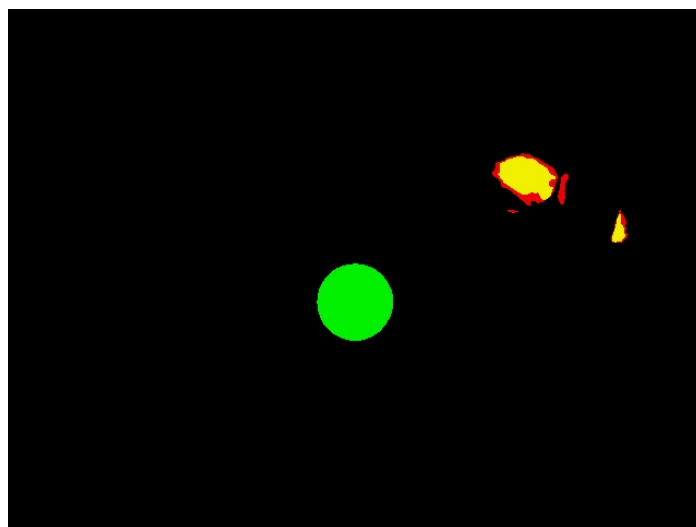


Figura 3.19: Imagem após o filtro de Threshold

É depois aplicado um filtro RGB para filtrar todas as cores menos o verde do comando:

```
dst = Ogre_Game::cvFilterRGB(frame, 0,50, 50,255 , 0,50, 0);
```

É usada a imagem a cores "frame" e depois de atribuídos os valores de cores que devem ser cortadas para se obter só a cor pretendida o resultado final é uma imagem com duas cores (preto e branco) em que o branco corresponde ao tom verde semelhante ao da bola do comando e o preto a todo o resto da imagem (ver Figura 3.20).

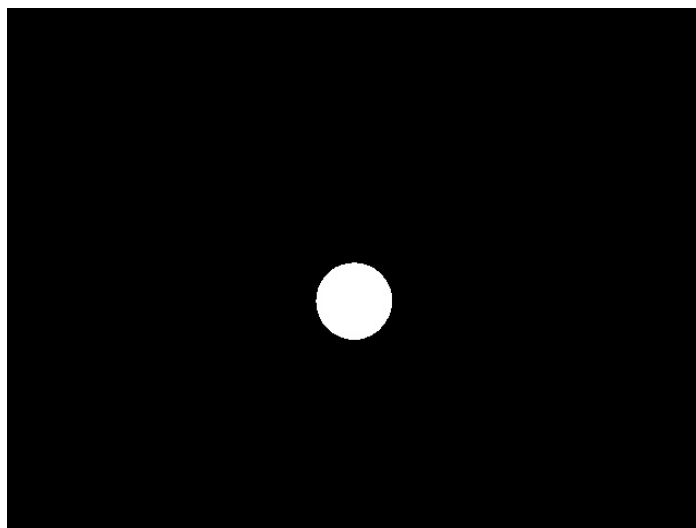


Figura 3.20: Imagem após todos os filtros

Temos agora uma imagem em que os pixéis brancos correspondem à bola do comando e os pixéis pretos correspondem ao resto o que nos permite calcular a posição do comando. Este cálculo consiste em dois ciclos, um ciclo que corre as linhas e outro ciclo dentro desse que corre as colunas. Para cada posição vai ser verificado se o pixel é preto ou branco, se é branco este pixel está na bola e os valores da posição actual são guardados como máximos. Também é guardada a diferença máxima entre pixéis brancos que estejam na mesma linha, resultando daí o diâmetro da bola (ver Figura 3.21).

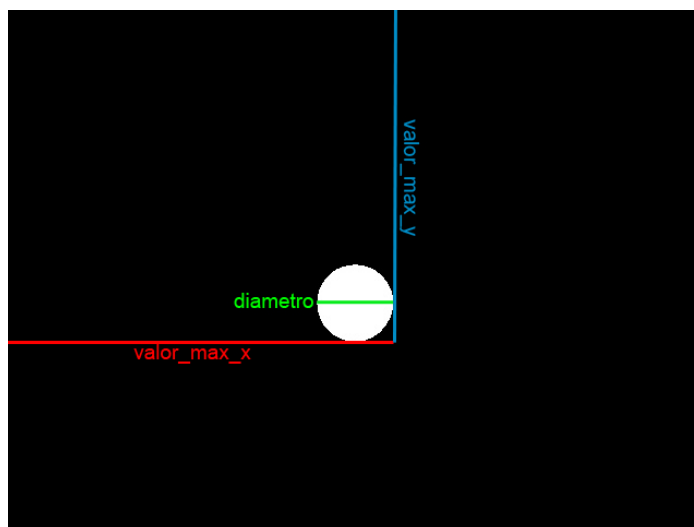


Figura 3.21: Calculo do diâmetro e dos valores máximos de X e de Y

No final iremos obter os valores máximos das posições e o diâmetro, e com estes valores podemos calcular o centro da bola, subtraindo o raio (diâmetro/2) aos valores máximos de X e de Y (ver Figura 3.22).

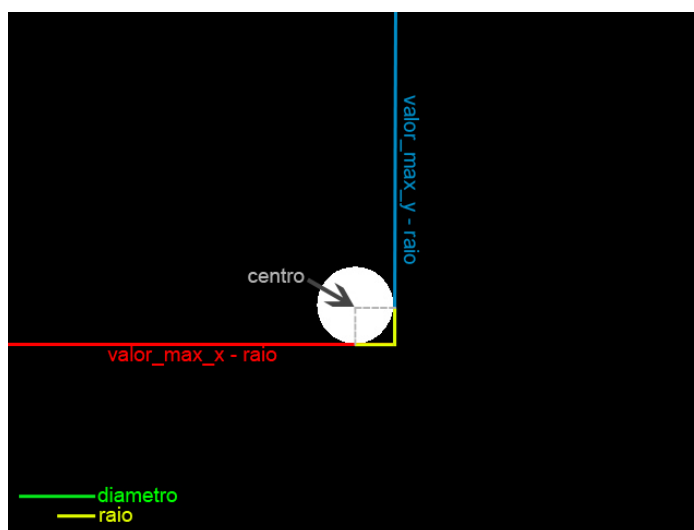


Figura 3.22: Calculo da posição central da bola

É depois efectuado um ajustamento dos valores das posições de acordo com a resolução da câmara (sendo neste caso 320*240) para que os valores do posicionamento tenham o ponto [0,0] como o centro do ecrã.

- $pos_x = pos_x - (diâmetro/2) - 320;$

- $pos_y = pos_y - (diâmetro/2) - 240;$

É também calculada a distância da bola à câmara (posição em Z) usando o logaritmo do diâmetro para tornar o valor exponencial do diâmetro mais linear.

- $pos_z = ((\log(diâmetro)))$;

Foram calculados assim os valores que permitem definir o posicionamento do comando no espaço, sendo estes usados na aplicação para poder simular o comportamento da raquete real.

Capítulo 4

Testes e resultados

4.1 Testes

Para se poder avaliar a interacção com o comando Move no jogo foi preciso efectuar testes com vários jogadores e capturar os seus resultados. Foram para isso realizadas duas sessões de jogo, uma sessão com alunos de Ciências do Desporto no pólo de Desporto da Universidade da Beira Interior com a ajuda da professora Kelly O'Hara e outra sessão na escola da Quinta das Palmeiras aos alunos com idades entre os 11 e os 15 com a orientação da professora Marta André (ver Figura 4.1). Nessas duas sessões foram usados dois computadores portáteis e dois conjuntos de acessórios Playstation Move + Câmara PSEye, isto para se poder obter o máximo de dados possíveis em menos tempo ao mesmo tempo que se aliciava os jogadores a competirem entre eles para obterem melhores resultados. Para se conseguirem dados com uma maior variedade de jogadores (a nível de idades e de preparação física) foi pedido a várias pessoas conhecidas que dessem o seu contributo para o estudo jogando também o jogo mSquash.

Em cada sessão foi dada a hipótese de o jogador usar o modo treino durante 5 minutos para se poder habituar minimamente à jogabilidade do mSquash. Após esse tempo o jogador introduzia os seus dados e jogava 15 bolas no modo de jogo Normal, e depois jogava outras 15 bolas no modo Target. Para poder analisar os dados de cada jogador foram usados dois métodos: foi gerado um relatório final de sessão para cada tipo jogo de acordo com os dados e performance do jogador e foi pedido ao jogador para preencher um questionário após ter jogado.



Figura 4.1: Sessão de jogo na Escola Quinta das Palmeiras

O relatório é gerado automaticamente no fim de cada sessão sendo adicionado ao ficheiro "relatório.txt" que contém as informações introduzidas pelo utilizador no início de cada sessão e os dados obtidos através da prestação do jogador ao longo do jogo. O relatório contém assim os seguintes dados:

- Id, Idade, Sexo e Mão Preferida do jogador
- Tipo de jogo (Normal ou Target)
- Número de bolas jogadas
- Tempo gasto na sessão de jogo
- Pontuação total, média e máxima numa bola
- Pontuação por bola (modo Normal)
- Número de bolas pontuadas por zona:
 - * Cima, Meio e Fundo para o modo normal
 - * Centro, 2º círculo e Círculo de fora para o modo target
- Número de bolas falhadas por zona: Direita, Esquerda, Baixo e Cima

Para poder avaliar a opinião do utilizador foi-lhe pedido para preencher um questionário em papel no final da sessão de jogo. Nos questionários foi usada a Escala Likert, criada por Rensis Likert[24] e que permite avaliar a opinião do utilizador sobre um certo tema/pergunta usando níveis (neste caso 5). Isto permite que seja mais fácil responder para o utilizador e mais fácil de analisar a sua opinião posteriormente.

Este questionário consiste em 4 partes: Avaliação do Jogo, Nível de Actividade Física do Jogador, Avaliação da Tecnologia e os Comentários globais/Sugestões.

Na avaliação do jogo foi pedido ao jogador para avaliar o nível de dificuldade, a vontade de voltar a jogar o jogo, o nível de actividade física, os efeitos sonoros, o aspecto gráfico, a ergonomia do comando de movimento e se aconselhava o jogo.

No nível de actividade física era questionado se o jogador costuma fazer desporto, se costuma jogar desportos que usem raquete e se está habituado a jogar videojogos que envolvam exercício físico (Exergames).

Na parte da avaliação da tecnologia foi pedido ao jogador para se avaliar como jogador de videojogos, para dar a opinião sobre jogos de computador, para avaliar o hábito de jogar com comandos de movimento, para definir o grau de interesse demonstrado por esta tecnologia e para dar a opinião sobre a utilidade deste tipo de jogos para o bem-estar e saúde dos jogadores.

Na última parte do questionário foi pedido ao jogador para identificar qual era o aspecto mais negativo no jogo, qual o mais positivo e as suas sugestões para o mesmo. Estes dados foram depois introduzidos e processados electronicamente usando uma folha de cálculo. De seguida são representados graficamente alguns dos resultados e são feitas interpretações sobre os mesmos.

4.2 Resultados

A população para o estudo foi obtida do seguinte modo: 23 indivíduos são estudantes do Departamento de Ciências do Desporto da UBI, 32 são alunos da Escola Secundária/3 Quinta das Palmeiras e os restantes 11 foram voluntários entre pessoas conhecidas. A população é assim constituída por 67 elementos que testaram o jogo mSquash e preencheram o questionário sobre o mesmo. Destas 67 pessoas, 45 eram do sexo masculino e 22 do sexo feminino(ver gráfico da Figura 4.2). A nível de mão preferida, 62 (93%) eram destros e 5 (7%) eram esquerdinos.



Figura 4.2: Gráfico da distribuição da população de acordo com o género

As idades dos jogadores variam entre os 11 e os 67 anos, e variam como o gráfico da Figura 4.3 nos mostra.

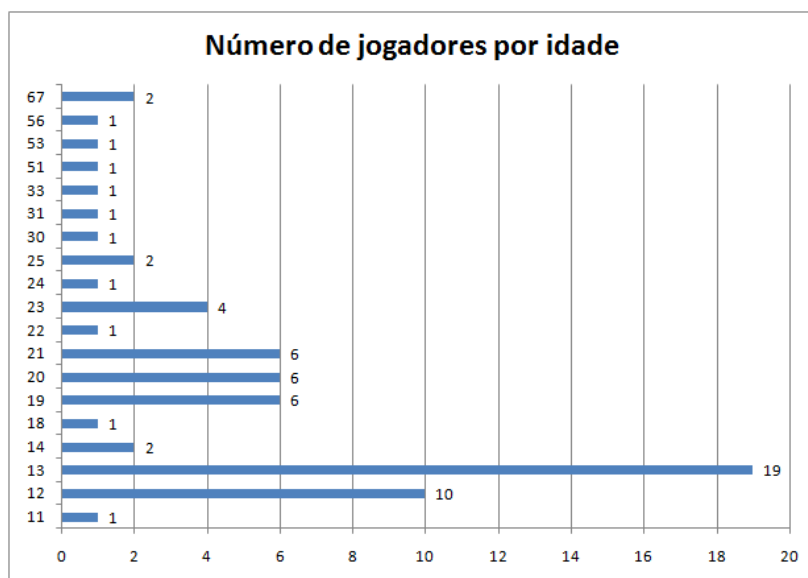


Figura 4.3: Gráfico da distribuição da população de acordo com a idade

Para comparar as pontuações de acordo com o género do jogador foi feita a média da pontuação final para cada género (ver gráfico da Figura 4.4). Verifica-se um equilíbrio nas pontuações no modo de jogo Normal e uma diferença do modo de jogo Target que favorece os jogadores do sexo masculino. Isto pode mostrar que os indivíduos do sexo

masculino se adaptam melhor ao jogo após apenas uns minutos bem como o facto de que os indivíduos do sexo feminino podem mostrar cansaço mais cedo e conseqüentemente prejudicar a sua prestação, pois a sessão no modo Target foi realizada a seguir à sessão no modo Normal.

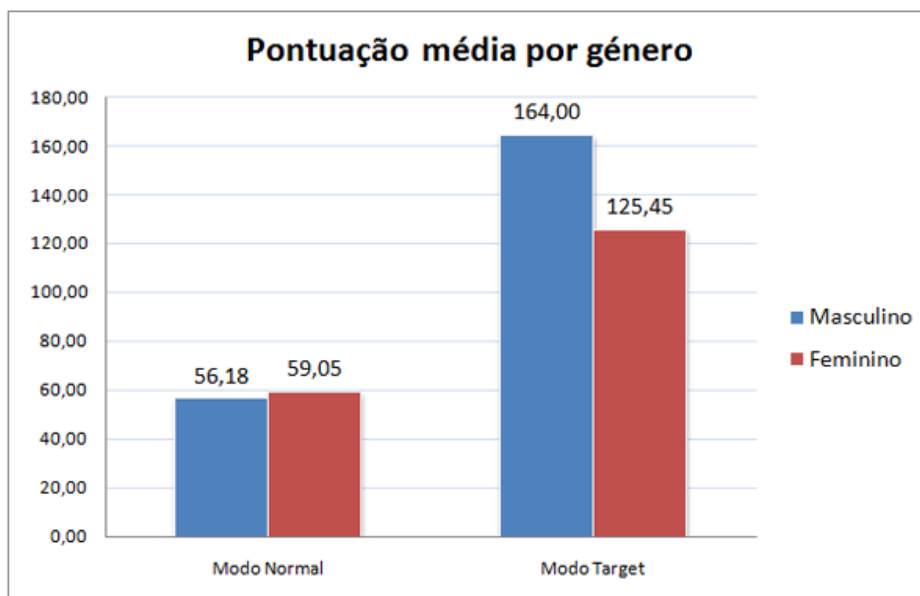


Figura 4.4: Gráfico da pontuação média final de acordo com o género do jogador

Para estudar o efeito do nível de dificuldade foi feita a média de pontuação por cada tipo de bola (ver gráfico da figura 4.5). As bolas fáceis tiveram uma pontuação média de 5,3 pontos, as bolas médias 3,8 e as bolas difíceis 2,3 pontos.

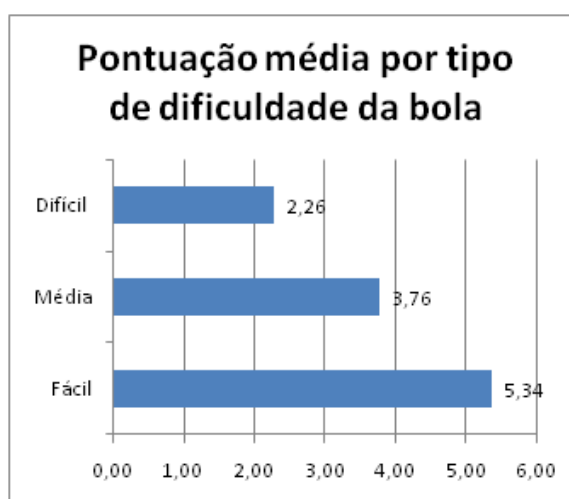


Figura 4.5: Gráfico da pontuação média em cada bola por nível de dificuldade

Como era pretendido com a criação do nível de dificuldade, diminuindo a área da raquete nota-se um decréscimo evidente na pontuação por bola ou seja, com o aumento do nível de dificuldade.

Para verificar a relação entre pontuação final e as idades dos jogadores foram criados 3 grupos: jogadores com menos de 16 anos (32), com idade entre os 16 e os 24 anos (25) e os jogadores com mais de 24 anos (10). As pontuações finais médias obtidas foram as seguintes(Figura 4.6):

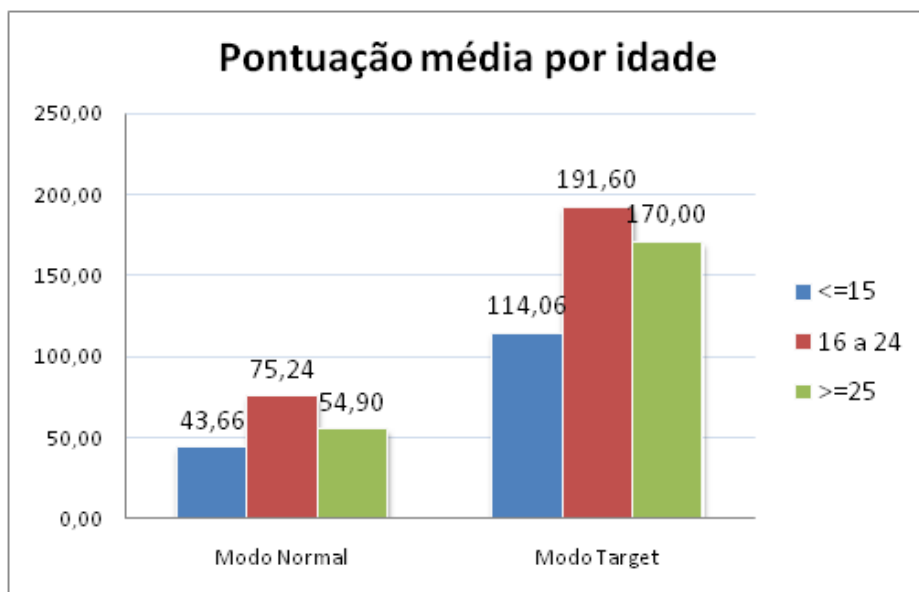


Figura 4.6: Gráfico da pontuação final média por idade

No modo normal há um certo equilíbrio entre a população mais jovem (<16) e a mais velha (>24), havendo uma vantagem do restante grupo (16 a 24). No segundo jogo há equilíbrio entre a população média e a mais velha, com os mais jovens a ficarem com menos pontuação média.

As pontuações mais baixas por parte dos jogadores mais jovens podem ser explicadas pelo facto destes serem menos experientes e talvez menos focados no objectivo, jogando mais por diversão pura. Os jogadores mais velhos têm pontuações ligeiramente mais baixas que os jogadores do grupo médio, isto pode-se dever ao facto de não serem tão ágeis como os jogadores mais jovens. O grupo de jogadores de idades compreendidas entre os 16 e os 24 mostra-se como o grupo melhor pontuado, por ter talvez o maior equilíbrio entre agilidade e experiência.

Para estudar o efeito do tempo de sessão na prestação dos jogadores foi calculado

o tempo médio por sessão para pontuações acima e abaixo da média de pontos para os dois modos de jogo. Foram obtidos os seguintes resultados (Figura 4.7):

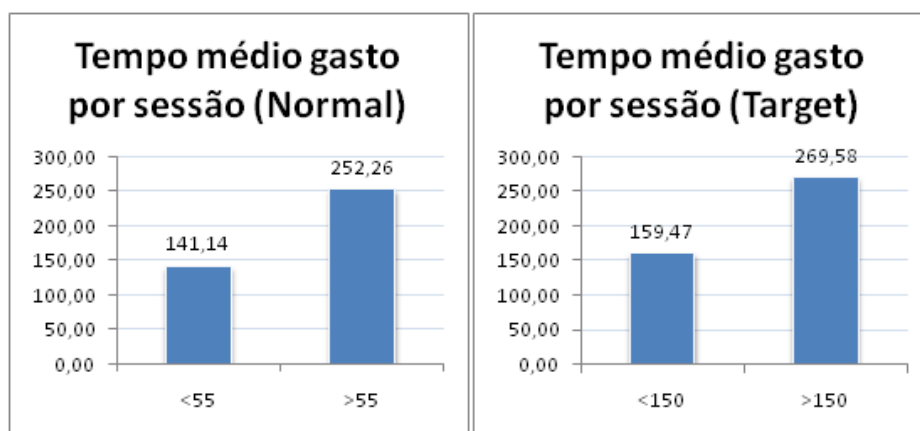


Figura 4.7: Gráfico do tempo de jogo médio gasto por sessão

Verifica-se tanto no modo normal como no modo target que para pontuações mais altas o tempo de sessão também é maior. Isto é natural, porque se um jogador jogar bem faz pontos e mantém a bola por mais tempo em jogo.

Foi calculada também a pontuação média das bolas por zona em que colidiu. No modo Normal temos 3 zonas: por cima das 2 linhas vermelhas, entre as linhas e por baixo das linhas. No modo Target temos o centro do alvo, a zona intermédia e o círculo exterior. As médias obtidas foram as seguintes(Figura 4.8):

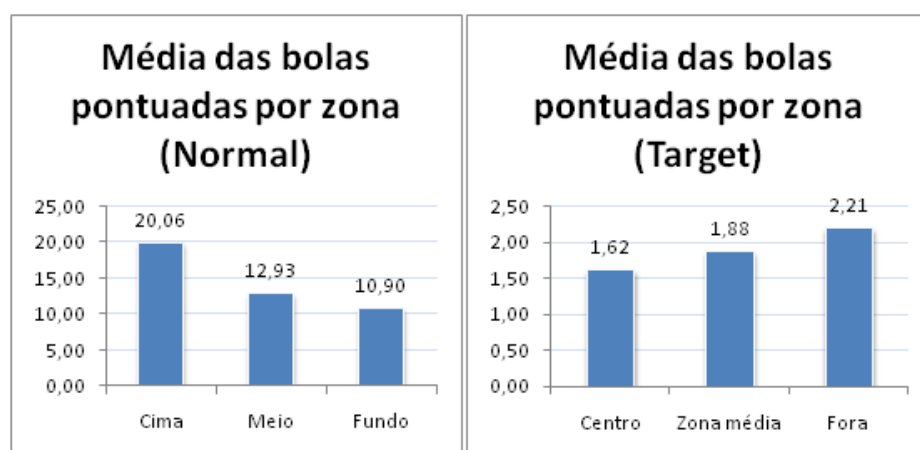


Figura 4.8: Gráfico do número de bolas pontuadas por zona

Como era pretendido, verifica-se que no modo Target a dificuldade aumenta com o aproximar da zona central do alvo, dando razão ao modo como foram atribuídos os

pontos. No modo normal observa-se que os jogadores pontuam mais vezes na zona de cima, o que pode ser sinal que jogam a bola com mais altura do que o pedido para a conseguirem manter mais tempo em jogo.

Foram registadas as zonas em que as bolas foram perdidas, dividindo o cenário horizontalmente e verticalmente. Foi possível também calcular o número de bolas que pararam a meio do cenário subtraindo ao total de bolas a soma das bolas perdidas por lados (direita+esquerda) ou por altura (baixo+cima). Isto permitiu calcular os valores médios do número de bolas perdidas por zona como se observa nos seguintes dados (ver Figura 4.9).

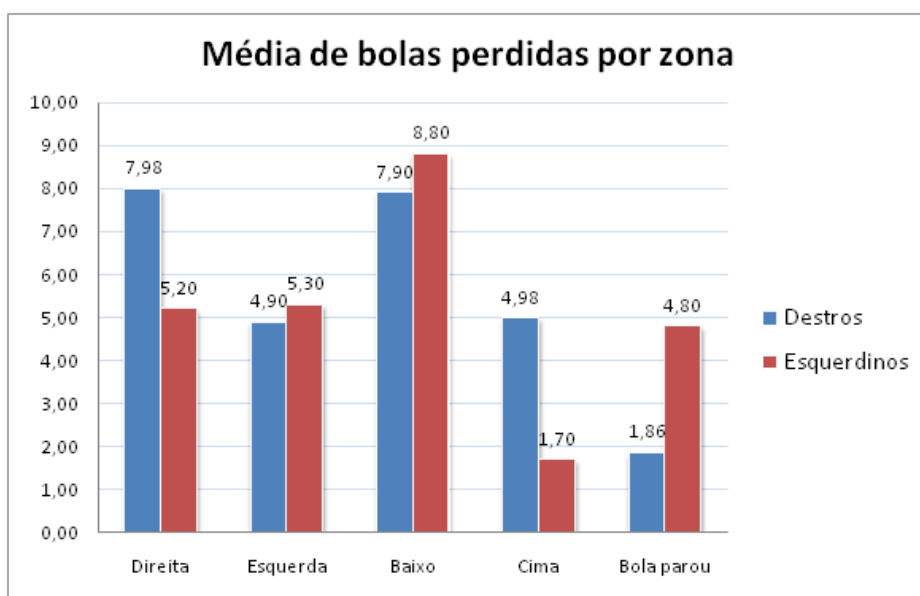


Figura 4.9: Gráfico do número de bolas perdidas por zona

Observa-se que os jogadores têm mais tendência em perder bolas pela parte inferior do cenário do que pela parte superior. Percebe-se também pelo gráfico que os jogadores esquerdinos perdem aproximadamente o mesmo número de bolas nos lados enquanto os jogadores destros têm tendência a deixar passar mais bolas pelo lado direito. Notar também que os jogadores esquerdinos deixam a bola parar mais vezes e perdem menos bolas por cima, sinal que poderão jogar com menos força que os jogadores destros.

Quando comparada a fluidez entre os dois portáteis usados nos testes reparou-se que um dos computadores (PC2) era mais lento na actualização de ecrã que o outro (PC1). Por isso foi registado para cada sessão qual o PC que tinha sido usado e esses

dados permitiram calcular a pontuação média por PC como mostra o gráfico da Figura 4.10.

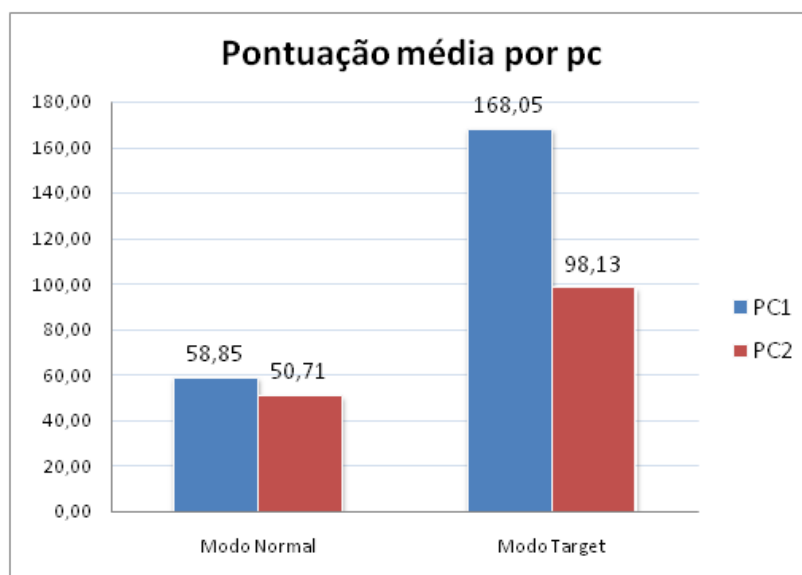


Figura 4.10: Gráfico da pontuação final média por PC

Nota-se que as pontuações do PC1 foram ligeiramente superiores no modo Normal e bastante superiores no modo Target. O que leva a concluir que uma menor fluidez afecta a jogabilidade do jogo de um modo visível.

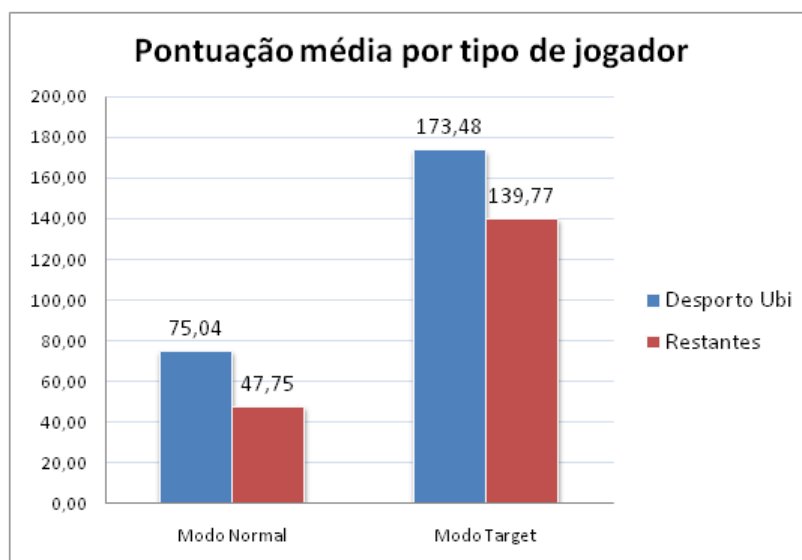


Figura 4.11: Gráfico da pontuação final média por tipo de jogador

Ao comparar as pontuações médias dos alunos do Departamento de Ciências do

Desporto da UBI com os restantes observou-se que tinham feito melhores resultados (gráfico da Figura 4.11).

Estes valores podem ser atribuídos ao facto dos alunos de Desporto terem uma maior facilidade de adaptação a um videojogo que envolva exercício físico (Exergame) devido aos diversos desportos que têm que aprender e também por serem mais preparadas fisicamente que os restantes.

Usar os valores das respostas dos questionários em conjunto com os dados dos relatórios permitiu fazer relações entre eles e chegar a algumas conclusões interessantes. De seguida é mostrado o gráfico que relaciona a pontuação final com o hábito de fazer desporto(Figura 4.12).

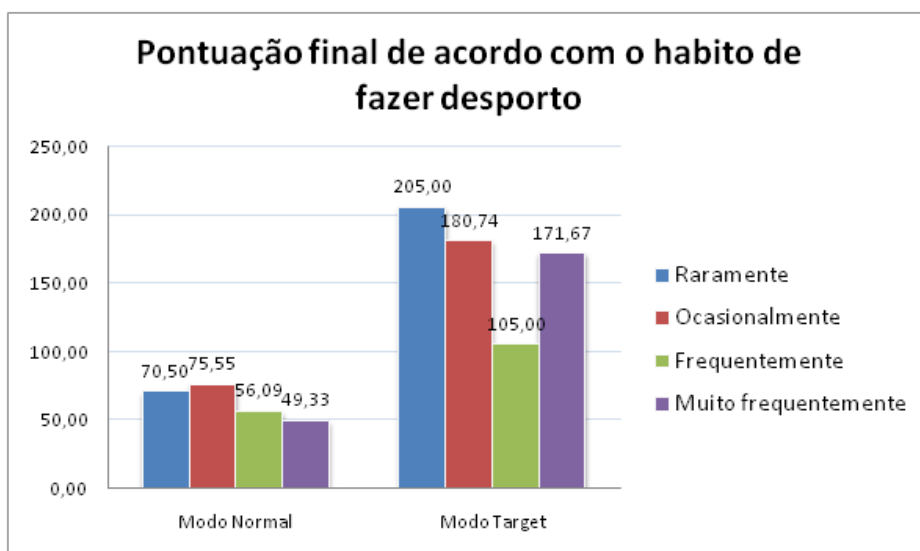


Figura 4.12: Gráfico da pontuação final média relacionado com o hábito de fazer desporto

Analisando o gráfico repara-se que não há qualquer associação entre o habito de fazer desporto e as pontuações obtidas. Isso pode ser considerado um factor positivo para o jogo, mostrando que qualquer pessoa, independentemente do nível físico, pode obter bons resultados. O que pode aliciar uma maior população de eventuais jogadores.

Foi também relacionada a média da pontuação final com o grau de dificuldade atribuído ao jogo por parte dos jogadores (Figura 4.13). Nota-se uma relação entre as pontuações obtidas e o nível de dificuldade avaliado pelo jogador. Os jogadores melhores pontuados foram os que acharam o jogo mais fácil, e os que tiveram piores pontuações os que o acharam muito difícil. (Notar que ninguém assinalou a opção 1 do questionário: "muito fácil").

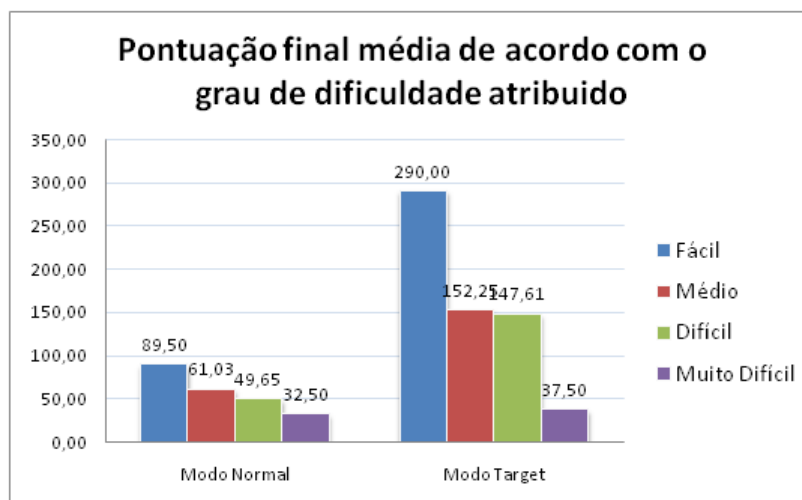


Figura 4.13: Gráfico da pontuação final média relacionado com o grau de dificuldade atribuído

Relacionou-se também a pontuação média final com o hábito do jogador de usar comandos de movimento em videojogos(Figura 4.14).

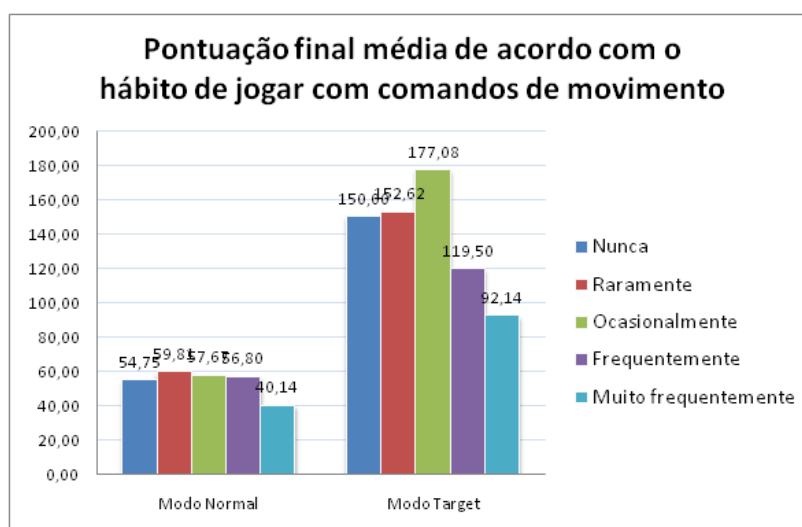


Figura 4.14: Gráfico da pontuação final média relacionado com o hábito de jogar com comandos de movimento

Tal como na relação entre a pontuação final e o hábito de fazer desporto, não se verifica qualquer padrão nesta relação. O facto dos jogadores habituados a usar comandos de movimento não terem obtido melhores pontuações mostra que o jogo não está semelhante aos jogos já existentes. Outro factor a ter em conta é o facto da maioria dos jogadores que responderam afirmativamente a esta questão dizerem que

tinham jogado com o comando da consola Wii, que difere do comando Move em muitos aspectos.

Foi depois relacionado o hábito de jogar desportos com raquete com a pontuação média final (gráfico da Figura 4.15). Tal como em exemplos anteriores, não foi observada nenhum padrão na relação entre estes dois valores. Isto mostra que o jogo mSquash pode estar um bocado longe da simulação que era pretendida, mas por outro lado que pode ser um jogo jogado sem experiência prévia em desportos que usem raquete.

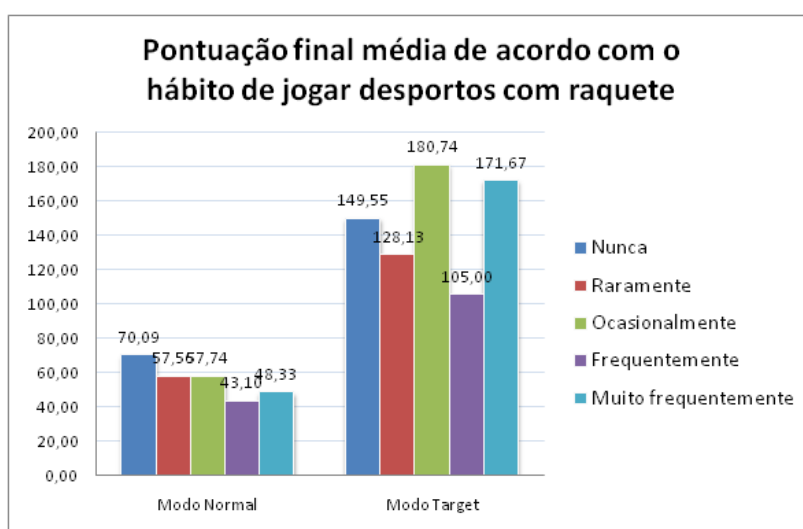


Figura 4.15: Gráfico da pontuação final média relacionado com o hábito de jogar desportos com raquete

Com as médias dos dados obtidos nos questionários é possível avaliar o jogo de acordo com a opinião dos jogadores. É possível também perceber a opinião dos jogadores sobre esta tecnologia e traçar um perfil geral desta população tanto a nível de actividade física como a nível do hábito de jogar videojogos.

Como se observa na tabela 4.1, o jogo foi avaliado positivamente, tendo recebido valores superiores a 3 em todas as categorias. Foi considerado um jogo com nível de dificuldade entre o Médio e o Difícil, mostrando que a curva de aprendizagem do jogo pode ser algo elevada. Foi também avaliado como sendo um jogo com um bom nível de actividade física, sendo este um dos objectivos de um Exergame. A nível gráfico e de som foi considerado Bom pelos jogadores.

Gráficos	3,87
Som	4,07
Dificuldade	3,33
Nível de actividade	3,73
Comando Move	4,27
Vontade de repetir	4,22
Aconselhava o jogo	4,12

Tabela 4.1: Pontuações do jogo atribuídas nos questionários

O comando de movimento Playstation Move recebeu também uma avaliação positiva sendo considerado Confortável por parte dos jogadores.

Por fim, é de realçar o facto dos jogadores terem avaliado positivamente a sua vontade de voltar a jogar e de aconselhar o jogo a alguém, sinal que o jogo mSquash teve um impacto positivo entre as pessoas avaliadas.

Segundo os dados obtidos no questionário (ver Tabela 4.2), no geral os jogadores avaliados são indivíduos que praticam actividades físicas, que gostam de videojogos e interessam-se por esta tecnologia. Notar também que os jogares não têm muito o hábito de jogar desportos com raquete e videojogos com comandos de movimento, podendo ser esta uma causa para um maior tempo de adaptação ao jogo.

Costumam fazer desporto	4,16
Costumam fazer desporto com raquete	2,67
Costumam jogar com comandos de movimento	2,56
Considera-se um jogador de videojogos	3,23
Opinião sobre videojogos	4,25
Grau de interesse por esta tecnologia	4,12
Acha que a tecnologia pode ser saudável	3,82

Tabela 4.2: Características dos jogadores retiradas das respostas dos questionários

Quando questionados sobre o facto desta tecnologia poder ser saudável para os jogadores responderam que pode ser útil com uma pontuação média de 3,8 em 5 pontos possíveis. Notar o facto de os alunos de desporto terem respondido com um valor

muito inferior nos questionários que os restantes jogadores (ver gráfico da Figura 4.16). Depois de ter ouvido a opinião de vários alunos deste grupo percebe-se bem o porquê desta diferença de opinião, pois os alunos consideram que é prejudicial para as suas futuras profissões que as pessoas façam o desporto em casa usando este tipo de tecnologia em vez de recorrerem aos ginásios e aulas desportivas. Outro factor que eles referiram foi precisamente o inverso da questão, ou seja, de o jogo poder ser prejudicial para o jogador, podendo causar lesões por estar a obrigar o jogador a fazer movimentos para acertar numa bola sem que haja uma força de resposta.

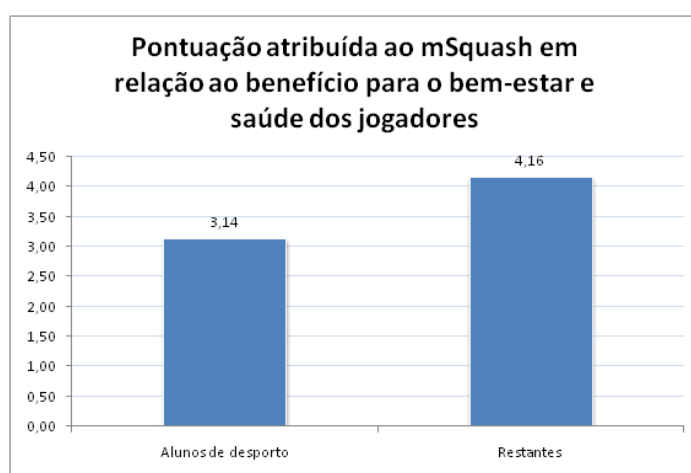


Figura 4.16: Gráfico da pontuação atribuída ao mSquash em relação ao benefício para o bem-estar e saúde dos jogadores

A tabela seguinte (Tabela 4.3) mostra os aspectos encontrados no jogo que foram considerados mais positivos pelos jogadores e o número de vezes que foram referidos.

Exercício físico (saudável)	15
Som do jogo	9
A interacção do jogo	9
Diversão	8
Realidade da simulação	6
Variedade do jogo	5
Aspecto gráfico	5

Tabela 4.3: Aspectos positivos do jogo mSquash apontados pelos jogadores testados

O aspecto considerado mais positivo foi o facto de o jogo ser um jogo saudável que promove o exercício físico. Os som, a interacção e a diversão foram outros factores considerados mais-valias do jogo mSquash.

Já os aspectos do jogo considerados mais positivos foram os seguintes(ver Tabela 4.4).

Orientação da raquete	6
Noção da distância e velocidade da bola	6
Controle da direcção bola	5
Dificuldade do jogo	4
Som do jogo	3
Aspecto gráfico	3
Curva de aprendizagem elevada	3

Tabela 4.4: Aspectos negativos do jogo mSquash apontados pelos jogadores testados

Os aspectos mais referidos pela negativa foram então a falta de orientação visual por parte da raquete, a dificuldade em perceber a distância e a velocidade da bola e o controle da bola quando colide com a raquete.

Reparar que houve aspectos que foram considerados positivos por uns jogadores e negativos por outros, como o aspecto gráfico e o som do jogo. A nível do som, houve jogadores que referiram ter gostado dos efeitos sonoros do publico no jogo enquanto outros referiram que era irritante. Em relação ao aspecto gráfico pode ser só uma questão de gosto, uns jogadores podem gostar do aspecto minimalista do jogo enquanto outros esperavam um cenário mais evoluído, ou até mesmo um corpo virtual para representar o jogador.

As sugestões dos jogadores serão discutidas no último capítulo, na secção do trabalho futuro.

Capítulo 5

Conclusões e trabalho futuro

Com a criação do jogo mSquash foi possível testar a interação com o jogo usando o comando de movimento Playstation Move, e avaliar as suas potencialidades. Foi ainda possível avaliar a opinião dos jogadores envolvidos nos testes realizados com o jogo, sobre este tipo de tecnologias que permitem mapear os movimentos do jogador para o jogo, e ao mesmo tempo efectuar exercício físico enquanto se joga

O motor de render OGRE mostrou ser uma excelente plataforma para o desenvolvimento de um videojogo, sendo um motor com um potencial enorme e em evolução constante.

Verificou-se também que a qualidade do hardware usado para o jogo pode influenciar muito a experiência de jogo e o nível de imersão do jogador, visto ter havido uma diferença de pontuações considerável entre os dois computadores usados, que tinham hardware com características diferentes, o que influenciava a fluidez de jogo.

Em relação ao comando Move, alguns jogadores ficaram admirados por verem este comando aplicado a um sistema diferente da consola Playstation 3. A maioria das pessoas afirmou ter gostado de experimentar esta tecnologia e muitos consideraram-na superior à encontrada no comando Wii da Nintendo.

Verificou-se também que os jogadores mais novos se mostravam mais entusiasmados com o jogo do que os mais velhos, havendo vários que exprimiram a sua vontade de repetir a experiência, e alguns ficaram mesmo à espera de uma nova vaga para poderem jogar, o que é um indicador positivo para o grau de entretenimento do jogo.

Houve jogadores em que se notou uma grande evolução durante o jogo de treino, e que se notou quando passaram para as versões a pontuar do mSquash, e outros que

não mostraram grande progressão na qualidade de jogo durante os 5 minutos de treino. Isso mostra que a facilidade de adaptação a uma nova realidade varia de pessoa para pessoa.

Com base nos dados obtidos nos testes realizados foi possível tirar algumas conclusões, como por exemplo, o efeito do grau de dificuldade nas prestações dos jogadores ou a relação entre as pontuações dos jogadores mais preparados fisicamente (alunos de desporto) e os restantes. O facto de as pontuações variarem segundo um mesmo padrão em relação à faixa etária para os dois tipos de jogo, que o tempo de sessão é inversamente proporcional à pontuação obtida e que os jogadores têm tendência para jogar com altura elevada quando querem manter a bola mais tempo em jogo.

Quanto à opinião dos jogadores, a grande maioria pontuou o jogo mSquash com pontuações positivas, sendo o comando Move e a vontade de repetir os itens com melhor pontuação. Os jogadores mostraram ser, na maioria, pessoas habituadas a fazer desporto e com um elevado grau de interesse pelos videojogos em geral. Apesar de muitos jogadores terem afirmado que não estão habituados a usar esta tecnologia de comandos de movimento, muitos mostraram uma rápida habituação e pontuaram esta tecnologia como uma tecnologia interessante.

Em relação aos aspectos que consideraram mais positivos no jogo, é agradável ver que o aspecto que os jogadores referenciaram mais vezes foi o facto do jogo mSquash ser um jogo saudável ao promover o exercício físico, o principal objectivo de um Exergame. Por isso podemos dizer que o comando Move adequa-se ao desenvolvimento deste tipo de jogos, como seria de esperar.

Em relação aos aspectos negativos, a orientação da raquete foi o principal aspecto realçado, seguindo-se a noção de distância e velocidade da bola. O problema da noção de distância e velocidade pode-se atribuir ao facto de o jogo ter que ser jogado a uma certa distância da câmara e do monitor do computador (aprox. 2m). Esta poderá ter sido uma distância muito grande para os ecrãs de 15 polegadas que faziam parte dos portáteis usados nos testes. Em relação à orientação da raquete, terão que ser testadas novas abordagens para obter um efeito mais realista. Outro factor referido como negativo foi a dificuldade da orientação da bola na colisão com a raquete.

Em relação às sugestões apresentadas, a resposta mais dada pelos jogadores foi que se continuasse a desenvolver o jogo pelo benefício que ele trazia à saúde dos jogadores. Outros jogadores referiram que seria interessante para o jogo ter mais modos de jogo,

que fosse criada a opção multi jogador ou contra computador. Estas sugestões são um bom ponto de partida para continuar a evolução deste videojogo, que é o pretendido.

Para concluir, pode-se dizer que a combinação dos Exergames com o tipo de interacção oferecida pelo comando Move mostrou que estas tecnologias são uma excelente solução para ajudar a resolver os problemas de saúde que se verificam na sociedade actual, pelo modo como conseguem cativar pessoas a jogar e, conseqüentemente, a praticarem exercício físico.

Como trabalho futuro, o ponto principal a desenvolver será a versão para dois jogadores, ou seja, a simulação de um jogo de squash. Este desenvolvimento é possível até pelo facto de o comando Move poder emitir várias cores, e assim basta ter uma cor diferente para cada um dos jogadores. Além disso, a parte da orientação da raquete precisa ainda de ser melhorada, bem como, a física e a colisão com a raquete. Resumindo, podemos dizer que o mSquash tem uma ampla margem de evolução.

Referências

- [1] Atari football. http://www.patrickscottpatterson.com/yahoo_site_admin/assets/images/AtariFootballProfile.16111803_large.jpg.
- [2] Pole position. http://www.pinballrebel.com/arcade/atari/pole_position/pole_position_b.jpg.
- [3] Pong iv. http://media.giantbomb.com/uploads/2/21593/817312-pong_iv_super.jpg.
- [4] Wikipedia - squash. http://pt.wikipedia.org/wiki/Ficheiro:Squash_court.JPG.
- [5] Original atari pong arcade console, 2007. <http://retrothing.typepad.com/photos/uncategorized/2007/09/12/ataripongconsole.jpg>.
- [6] B. Bailey and K. McInnis. Energy cost of exergaming: A comparison of the energy cost of 6 forms of exergaming. *Archives of pediatrics and adolescent medicine*, 2011.
- [7] I. Bogost. The rhetoric of exergaming. *Digital Arts and Cultures conference*, 2005.
- [8] D. Bos, B. Reuderink, B. Laar, H. Gurkok, C. Muhl, M. Poel, D. Heylen, and A. Nijholt. Human-computer interaction for bci games. *2010 International Conference of Cyberworlds*, 2010.
- [9] X. Cai. Principles of human-computer interaction in game design. *2009 Second International Symposium on Computational Intelligence and Design*, 2009.
- [10] M. Casamassina. E3 2007: Nintendo e3 media briefing live blog. <http://uk.wii.ign.com/articles/803/803335p1.html>.

- [11] C. Crawford. *Chris Crawford on Game Design*. New Riders, June 2003.
- [12] M. Csikszentmihalyi. *Beyond Boredom and Anxiety: Experiencing Flow in Work and Play*. Jossey-Bass, 1975.
- [13] Dead Mage. Garshasp. <http://garshasp.com/>.
- [14] EA Sports. Ea sports active 2. <http://www.easportsactiveonline.com/fitness-games/ea-sports-active-2>.
- [15] B. Edwards. Video games turn 40, 2007. <http://www.1up.com/features/videogames-turn-40?pager.offset=0>.
- [16] Eurogamer. Move sales top 8.8m. <http://www.eurogamer.net/articles/2011-06-07-move-sales-top-8-8m>.
- [17] S. Finkelstein, A. Nickel, T. Barnes, and E. Suma. Astrojumper: Designing a virtual reality exergame to motivate children with autism to exercise. *IEEE Virtual Reality 2010*, 2010.
- [18] Firelight Technologies. fmod: interactive audio middleware. <http://www.fmod.org/>.
- [19] Global Health Observatory. Portugal: health profile. Technical report, World Health Organization, 2011.
- [20] P. Hamalainen, J. Hoysniemi, T. Rouvi, and L. Turkki. Quiqui's giant bounce, 2003. http://www.cs.uta.fi/kukakumma/htmls_en/index.html.
- [21] J. Hoysniemi. *Design and Evaluation of Physically Interactive Games*. PhD thesis, University of Tampere, 2006.
- [22] Konami. Dance dance revolution, 2010. <http://www.konami.com/officialsites/ddr2010/>.
- [23] S. Lawrence. Exercise, lose weight with exergaming, 2005. <http://www.foxnews.com/story/0,2933,144761,00.html>.
- [24] R. Likert. *A technique for the measurement of attitudes*. Archives of psychology n.140, 1932.

- [25] A. P. Macvean. Task-involved versus ego-involved: Motivating children to exercise in a pervasive exergame. *Fourth Annual PhD Forum on Pervasive Computing and Communications*, 2011.
- [26] Metacritic. Ankh - metacritic. <http://www.metacritic.com/game/pc/ankh>.
- [27] Metacritic. Garshasp: The monster slayer - metacritic. <http://www.metacritic.com/game/pc/garshasp-the-monster-slayer>.
- [28] Metacritic. Hardware review: Playstation move. <http://www.metacritic.com/feature/playstation-move-reviews>.
- [29] Metacritic. Nimbus - metacritic. <http://www.metacritic.com/game/pc/nimbus>.
- [30] Metacritic. Torchlight - metacritic. <http://www.metacritic.com/game/pc/torchlight>.
- [31] Metacritic. Under siege - metacritic. <http://www.metacritic.com/game/playstation-3/under-siege>.
- [32] Microsoft. Microsoft kinect, 2011. <http://www.xbox.com/pt-PT/kinect>.
- [33] moveonpc. Supporting the playstation move controller as an input device on the pc. <http://code.google.com/p/moveonpc/>.
- [34] Nintendo. Wii fit plus. <http://wiifit.com/>.
- [35] Nintendo. Iwata asks - wii sports resort, 2010. http://us.wii.com/iwata_asks/wiisportsresort/voll1_page4.jsp.
- [36] Nintendo. Supplementary information about earnings release, 2010. <http://www.nintendo.co.jp/ir/pdf/2010/100507e.pdf>.
- [37] Nintendo. Nintendo 3ds, 2011. <http://www.nintendo.com/3ds>.
- [38] Nintendo. Nintendo wii, 2011. <http://www.nintendo.com/wii>.
- [39] Noumenon Games. Nimbus. <http://noumenongames.com/>.

- [40] Ogre Wiki. Basic ogre framework. <http://www.ogre3d.org/tikiwiki/tiki-index.php?page=Basic+Ogre+Framework>.
- [41] C. Onyett. Nimbus review, 2010. <http://uk.pc.ign.com/articles/113/1132580p1.html>.
- [42] OpenCV. Opencvwiki. <http://opencv.willowgarage.com/wiki/>.
- [43] Z. Pan, W. Xu, J. Huang, M. Zhang, and J. Shi. Easybowling: a small bowling machine based on virtual simulation. *Computers and Graphics* 27, 2003.
- [44] Prime Games. Ankh. <http://www.ankh-game.com/>.
- [45] D. F. Roberts. *Kids and media @ the new millennium*. Kaiser Family Foundation, 1999.
- [46] D. Robinson, I. Palmer, P. Excell, R. Earnshaw, and O. Salem. Multi-platform human computer interaction in converged media spaces. *2009 International Conference on CyberWorlds*, 2009.
- [47] Runic Games. Torchlight. <http://www.torchlightgame.com/>.
- [48] N. Santelmann. Ten healthiest sports. *Forbes Magazine*, 2003. http://www.forbes.com/2003/10/01/cx_ns_1001feat.html.
- [49] Seed Studios. Official under siege website, 2011. <http://www.undersiegegame.com/>.
- [50] J. Sinclair, P. Hingston, and M. Masek. Considerations for the design of exergames. *GRAPHITE '07 Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, 2007.
- [51] J. Sinclair, P. Hingston, and M. Masek. Testing an exergame for effectiveness and attractiveness. *2010 2nd International IEEE Consumer Electronics Society's Games Innovations Conference*, 2010.
- [52] J. Sinclair, P. Hingston, M. Masek, and K. Nosaka. Using a virtual body to aid in exergaming system development. *IEEE Computer Graphics and Applications*, 2009.

- [53] G. Sinha, R. Shahi, and M. Shankar. Human computer interaction. *Third International Conference on Emerging Trends in Engineering and Technology*, 2010.
- [54] Sony Playstation. Comando sem fios dualshock®3. <http://pt.playstation.com/ps3/peripherals/detail/item114225/Comando-sem-fios-DUALSHOCK%C2%AE3/>.
- [55] Sony Playstation. Playstation eye. <http://pt.playstation.com/ps3/peripherals/detail/item78905/PlayStation%C2%AEEye/>.
- [56] Sony Playstation. Playstation move sharp shooter. <http://us.playstation.com/ps3/accessories/playstation-move-sharp-shooter-ps3.html>.
- [57] Sony Playstation. EyeToy: Kinetic, 2005. <http://pt.playstation.com/ps2/games/detail/item42723/EyeToy-Kinetic/>.
- [58] Sony Playstation. Playstation move, 2011. <http://pt.playstation.com/psmove/>.
- [59] S. Streeting. Ogre manual v1.7('cthugha'), 2010. <http://www.ogre3d.org/docs/manual/index.html>.
- [60] F. Technologies. Fmod downloads. <http://www.fmod.org/index.php/download>.
- [61] The International Arcade Museum Library. Pong. http://www.arcade-museum.com/game_detail.php?game_id=9074.
- [62] Torus Knot Software. Ogre. <http://www.ogre3d.org/>.
- [63] United States Army. America's army, 2011. <http://www.americasarmy.com/>.
- [64] Wikipedia. Earth eternal. http://en.wikipedia.org/wiki/Earth_Eternal.
- [65] Wikipedia. Power pad. http://en.wikipedia.org/wiki/Power_Pad.

- [66] Y. Zhan and Y. Ding. *The Designing and Development of Game*. People's Traffic Press, 2003.
- [67] M. Zhang, M. Xu, Y. Liu, G. He, L. Han, P. Lv, and Y. Li. The framework and implementation of virtual network marathon. *IEEE International Symposium on Virtual Reality Innovation 2011*, 2011.

Anexo: Instalação do jogo mSquash

Instalação do comando Move

1. Requisitos

- Comando Playstation Move
- Adaptador Bluetooth
- Cabo mini-USB
- Software MotionInJoy

2. Instalar o software de controlo do comando e as drivers

- Fazer download do pacote MotionInJoy_060001 correspondente ao sistema operativo no site:
 - (a) 32 bits Windows xp,Windows vista,Windows 7:
http://www.xinputer.com/download/MotionInJoy_060001_x86_signed.zip
 - (b) 64 bits Windows vista,Windows 7:
http://www.xinputer.com/download/MotionInJoy_060001_amd64_signed.zip
- Extrair o ficheiro .rar, correr o executável e instalar o software.
- Como a ferramenta "DS3 Tool" corre com ligação à Internet é preciso fazer o download e instalação de vários ficheiros, ficando assim a trabalhar em modo offline. Para isso é preciso fazer os seguinte downloads:
 - (a) <http://forums.motioninjoy.com/download/file.php?id=92>
 - (b) <http://forums.motioninjoy.com/download/file.php?id=93>
 - (c) <http://forums.motioninjoy.com/download/file.php?id=94>

Depois basta extrair a pasta "local" para a seguinte localização:

(a) "...\MotioninJoy\ds3"

- Executar a ferramenta "DS3 Tool" e ligar o comando Move e o adaptador bluetooth (ou desactivar o rádio bluetooth caso este já esteja instalado no sistema operativo).
- Carregar no separador "Gestão de Drivers" no topo e seleccionar a opção Load (ver Figura 1).
- Se tudo correu conforme planeado mostrará o adaptador Bluetooth na opção "Bluetooth Adapter" e o comando Move na opção "DUALSHOCK3", como mostra na figura 2.
- Carregar no botão "Pair Now" para conectar o comando ao pc via Bluetooth, desligar o comando do cabo mini-USB e pressionar o botão Playstation (botão redondo com um P). Se o led vermelho ligar com luz contínua é porque o comando está configurado e preparado para jogar.



Figura 1: Carregar drivers na ferramenta DS3 Tool



Figura 2: Conectar o comando por Bluetooth

Instalação da câmara PSEye

1. Requisitos

- Câmara Playstation Eye (PSEye)
- Drivers Code Laboratories

2. Instalar drivers e software de configuração da câmara PSEye

- Fazer download do instalador através do seguinte link:
 - (a) <http://codelaboratories.com/files/archive/CL-Eye-Driver-4.0.2.1017.exe>
- Correr o executável e seguir as instruções para instalar as drivers e o software de controlo da câmara (o instalador precisa de uma ligação à Internet).
- Após instalar o software, ligar a câmara ao pc por USB e correr o executável "CL-Eye Test". Escolher a opção Options -> Video Capture Filter... (ver Figura 3).

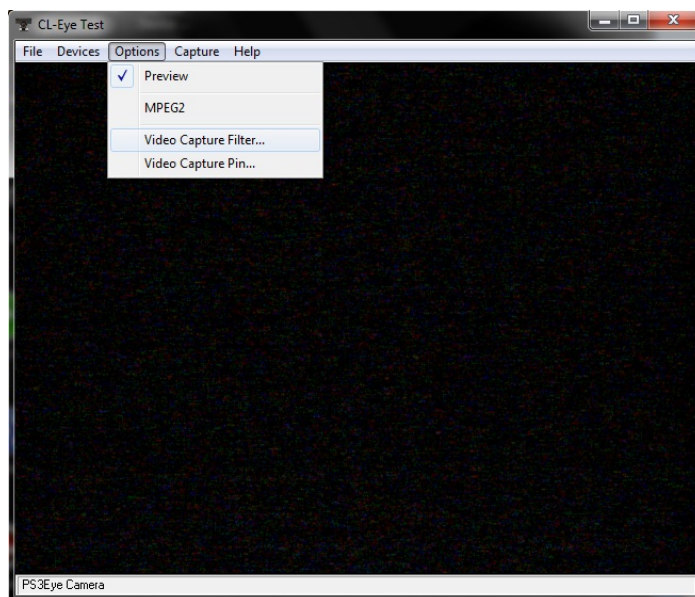


Figura 3: Configurar as opções da câmara

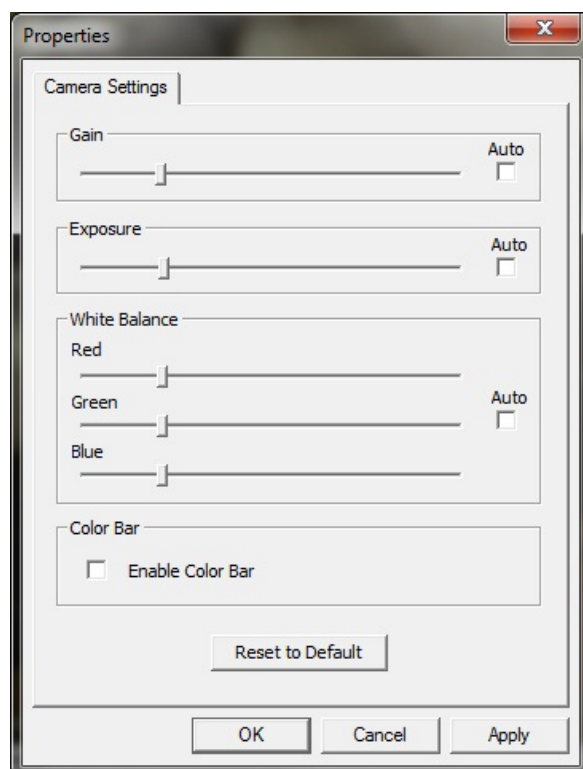


Figura 4: Ajuste dos parâmetros da câmara

- Desmarcar os quadrados com a opção Auto e dispor as barras como mostra na imagem seguinte (pode ter que haver um pequeno ajustamento de acordo

com o ambiente em que estamos a jogar)(ver Figura 4).

- Pressionar o botão OK e sair da aplicação para poder jogar. A câmara está agora instalada e pré-calibrada.

Configuração do jogo

1. Requisitos

- DirectX9:
<http://www.microsoft.com/downloads/details.aspx?FamilyID=2da43d38-db71-4c1b-bc6a-9b6652cd92a3&displayLang=pt-br>
- Windows Xp/Vista/Seven
- Comando Playstation Move
- Câmara Playstation Eye

2. Configurar a câmara e o espaço de jogo

- Ligar a câmara PSEye ao pc
- Apontar o PC e a câmara para uma zona que não tenha luz directa
- Correr o executável "tracking.exe"para verificar se a zona de jogo é uma zona boa para jogar
- Verifica-se que existem duas imagens, sendo uma a imagem real e a outra a imagem filtrada. O ideal será posicionar a câmara de modo a que na imagem filtrada não existam pixéis brancos (ver na Figura 5).

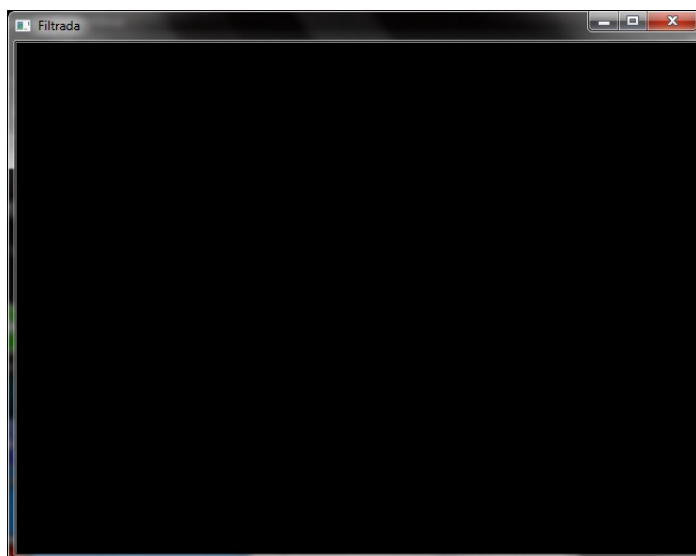


Figura 5: Espaço de jogo correctamente configurado

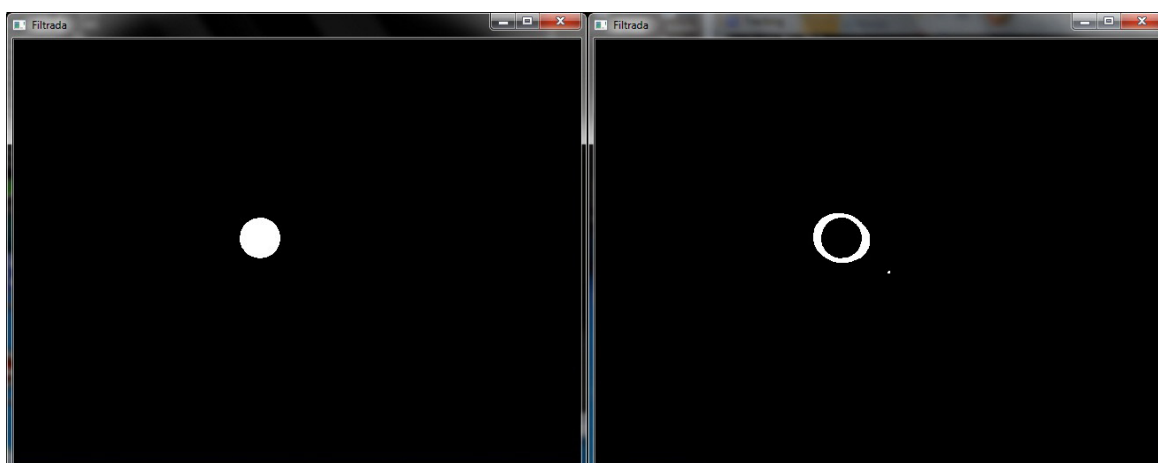


Figura 6: Imagem correctamente configurada (esquerda) e incorrectamente configurada (direita)

3. Configurar o tracking do comando

- Conectar o comando Move por bluetooth ao pc
- Executar a aplicação "tracking.exe" para iniciar o tracking e depois iniciar o executável "mudacorverde.exe" para ligar a luz do comando com a cor verde
- Verificar se na imagem filtrada aparece um círculo branco quase perfeito, caso apareça é sinal que está tudo configurado e preparado para jogar.
- Caso não apareça um círculo perfeito (lado direito da Figura 6) ou não apareça mesmo nada na imagem filtrada é preciso ajustar os parâmetros de

captura de imagem como foi feito no exemplo da figura 4 deste apêndice. Para isso tem que se fechar a aplicação "tracking.exe" em primeiro lugar. Após reajustar os parâmetros, desligar o "CLEye Test" e voltar à alínea b.

4. Iniciar o jogo

- Desligar as aplicações de configuração e correr o executável "mSquash.exe" para iniciar o jogo
- Usar qualquer tecla do teclado para passar os primeiros dois ecrãs iniciais
- Preencher o relatório com os dados pessoais (Figura 7).

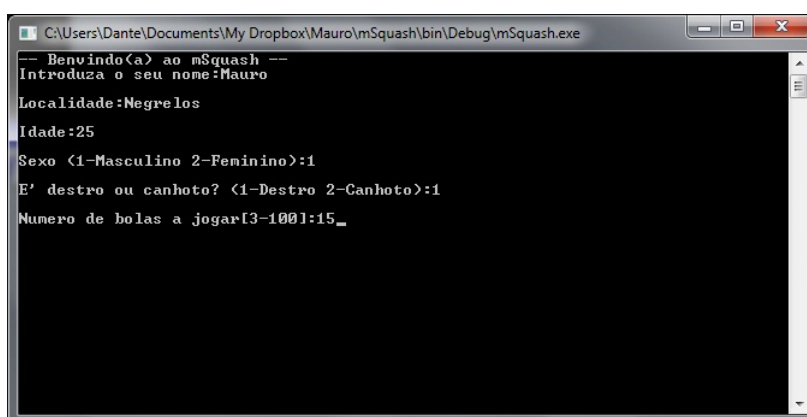


Figura 7: Introdução dos dados do utilizador

- É pedido depois ao utilizador para apontar o comando para o ecrã e pressionar o botão Move do comando, e depois são dadas instruções ao jogador para calibrar a distância ideal.
- Aparece o ecrã de selecção de jogo, em que podemos escolher o jogo no modo Normal (pressionar o botão Quadrado) ou o modo Target (pressionar o botão Triangulo) (ver Figura 8).



Figura 8: Ecrã de selecção do tipo de jogo

- O jogo está agora configurado e a correr (ver Figura 9).



Figura 9: Jogo a correr