



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Combinando Kinect e smartphones para rastreamento do movimento do pulso.

Alexandre Henrique Aben-Athar Sousa Baião
Diego Bichara de Lima

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Marcus Vinicius Lamar

Brasília
2014

Universidade de Brasília — UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação

Coordenador: Prof. Dr. Homero Luiz Picollo

Banca examinadora composta por:

Prof. Dr. Marcus Vinicius Lamar (Orientador) — CIC/UnB

Prof.^a Dr.^a Carla Denise Castanho — CIC/UnB

Prof.^a Dr.^a Carla Maria Chagas e Cavalcante Koike — CIC/UnB

CIP — Catalogação Internacional na Publicação

Aben-Athar Sousa Baião, Alexandre Henrique.

Combinando Kinect e smartphones para rastreamento do movimento do pulso. / Alexandre Henrique Aben-Athar Sousa Baião, Diego Bichara de Lima. Brasília : UnB, 2014.

111 p. : il. ; 29,5 cm.

Monografia (Graduação) — Universidade de Brasília, Brasília, 2014.

1. Kinect, 2. Jogos, 3. Acelerômetro, 4. Giroscópio, 5. Rastreamento,
6. Gestos, 7. Imersão virtual

CDU 004.4

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro — Asa Norte
CEP 70910-900
Brasília-DF — Brasil



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Combinando Kinect e smartphones para rastreamento do movimento do pulso.

Alexandre Henrique Aben-Athar Sousa Baião

Diego Bichara de Lima

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Marcus Vinicius Lamar (Orientador)

CIC/UnB

Prof.^a Dr.^a Carla Denise Castanho

CIC/UnB

Prof.^a Dr.^a Carla Maria Chagas e Cavalcante Koike

CIC/UnB

Prof. Dr. Homero Luiz Picollo

Coordenador do Bacharelado em Ciência da Computação

Brasília, 23 de julho de 2014

Dedicatória

Aos nossos pais, Gil Weber Baião, Jaqueline Aben Athar de Souza, Aristides Terêncio de Lima Filho e Maria Andréa Bichara de Lima, sem os quais este trabalho não seria possível e que nos inspiraram durante essa longa jornada que por diversas vezes pensamos em desistir.

Agradecimentos

Agradecemos ao Diego Pereira, Fernando Feliu Gonzales, Gabriel Bichara de Lima, Ivan Augusto Dantas, Lucas Bichara de Lima, Luiz Roberto Gomes Bichara, Mateus Sampaio, Sônia Maria da Cunha Bichara e Thaís Magalhães Maciel, cada um ajudou da forma que pôde na construção do trabalho e na vida dos que o fizeram. Ao nosso orientador Prof. Dr. Marcus Lamar pelo estímulo, solidariedade e presteza no auxílio às atividades deste trabalho. À Prof.^a Dr.^a Carla Castanho e à Prof.^a Dr.^a Carla Koike pelo carinho e disponibilidade em auxiliar a todos os alunos em suas dificuldades e também por participarem de nossa banca examinadora.

Abstract

As interfaces homem/máquina estão cada vez se tornando mais naturais, sendo o *Kinect*, *PlayStation Move* e *Wii Remote*, exemplos dessa evolução. Entretanto, tais interfaces têm suas limitações - o *Kinect*, por exemplo, não suporta um rastreamento de pulso eficiente; principalmente pela pouca precisão resultante do custo baixo que tal equipamento precisa ter para ser competitivo no mercado. Por outro lado, celulares *smartphones* já são comumente utilizados pelas pessoas em suas atividades diárias e a maioria destes dispositivos já incorpora vários tipos de sensores como um GPS, acelerômetro, giroscópio, etc. Este trabalho propõe o uso de um dispositivo móvel dotado de sensores acelerômetro e giroscópio, como um *smartphone*, para incrementar a adicionar o rastreamento dos movimentos do pulso em aplicações e jogos que utilizam o *Kinect*, a um custo relativamente baixo. O sistema é baseado no uso conjunto de uma biblioteca de desenvolvimento disponível para *Kinect* e comunicação com dispositivos móveis por via de uma rede *Wi-Fi*. Um jogo de luta com espadas para dois jogadores foi desenvolvido utilizando o sistema proposto e a plataforma de desenvolvimento de jogos *Unity3D*. São apresentados os resultados experimentais das medidas da precisão e acurácia do rastreamento do pulso dos jogadores. Uma pesquisa de opinião contando com 27 jogadores participantes indica que a experiência de jogo foi consideravelmente melhorada quando comparada a um sistema usando somente o sensor *Kinect*. Nem a acurácia limitada dos acelerômetros e giroscópios ou o atraso no tempo devido à conexão *wireless* foram experienciados pelos jogadores.

Palavras-chave: Kinect, Jogos, Acelerômetro, Giroscópio, Rastreamento, Gestos, Imersão virtual

Abstract

The interfaces between players and machines are rapidly becoming more natural, being Kinect, PlayStation Move and Wii Remote great examples. However, these interfaces still have many limitations - Kinect, as a case, lacks an efficient wrist tracking; mainly due to the poor accuracy resulting from the low cost that such equipment must have in order to be competitive in the market. Smartphones are already commonly used by people in their daily activities, and most of these devices already incorporate various kinds of sensors, such as a GPS, accelerometer, gyroscope, compass, etc. This work proposes the use of a portable device provided with accelerometer and gyroscope sensors, such as a smartphone, to add the efficient tracking of the wrist movements in applications and games that utilize Kinect, at a relative low cost. The system is based on the joint use of a development library available for Kinect and communication with mobile devices via Wi-Fi network. A two player sword fight game was developed using the proposed system and the Unity 3D framework. There are presented the experimental results of the measurement of the precision and accuracy of the tracking of the players' wrist. An opinion poll with 27 gamers indicates that the gameplay is considerably improved when compared to a system using only the Kinect sensor. Neither the limited accuracy of the accelerometers and gyroscopes nor the delay time of the wireless communication were experienced by the players.

Keywords: Kinect, Games, Accelerometer, Gyroscope, Tracking, Gesture, Virtual immersion

Sumário

1	Introdução	1
2	Referencial Teórico	5
2.1	Trabalhos Correlatos	5
2.1.1	Rastreamento da Mão	5
2.1.2	Rastreamento corporal	8
2.1.3	Trabalhos Semelhantes	11
2.2	Kinect	14
2.3	Giroscópio	15
2.4	Acelerômetro	16
2.5	Framework Unity	19
3	Implementação do Sistema	21
3.1	Captura do movimento do celular	21
3.2	Captura do movimento corporal	24
3.3	O Jogo	25
4	Resultados Obtidos	29
4.1	Resultados do rastreamento de pulso	29
4.2	Resultados do rastreamento corporal	30
4.3	Pesquisa de opinião	34
5	Conclusão	43
	Referências	44

Lista de Figuras

1.1	Infográfico de evolução dos controles de console da PopChartLab [24].	2
1.2	Controles modernos	3
1.3	Infográfico disponibilizado pela AG2 Publicis Modem [12]	4
2.1	Rastreamento utilizando duas câmeras [43]	6
2.2	Rastreamento da mão com interação com objeto [36]	6
2.3	Rastreamento da mão utilizando PSO [30]	6
2.4	Rastreamento da mão com interação de objeto [31]	7
2.5	Rastreamento da mão proposta em [26]	7
2.6	Utilização do algoritmo proposto em [41]	8
2.7	Rastreamento corporal utilizando [34].	8
2.8	Rastreamento por temperatura corporal [22]	9
2.9	Localização do sujeito dentro de uma sala [25]	9
2.10	Modelo gerado pela movimentação de um participante em [15].	10
2.11	Captura de quadros utilizando o <i>Ray-Constrained ICP Model</i> [16]	11
2.12	Sistema sendo utilizado em dois celulares para controle de um jogo de corrida <i>multiplayer</i> [28]	12
2.13	Framework sendo utilizado em um jogo funcional [21]	12
2.14	Reconhecimento de gestos em [40]	13
2.15	Demonstração do SwordPlay em execução [23]	13
2.16	Jogo <i>Ninja Gaiden 3</i> da <i>Team Ninja</i> [20]	14
2.17	Força <i>Coriolis</i> para um corpo [42].	16
2.18	Esquema de funcionamento de um giroscópio MEMS [42]	16
2.19	Exemplo de um acelerômetro capacitivo [17]	17
2.20	Exemplo de um acelerômetro piezoelétrico [9]	17
2.21	Exemplo de um acelerômetro piezoresistivo [29]	17
2.22	Exemplo de um acelerômetro utilizando o efeito Hall [33]	18
2.23	Exemplo de um acelerômetro Magnetoresistivo [1]	18
2.24	Exemplo de um acelerômetro usando transferência de calor [35]	18
2.25	Esquema de um acelerômetro capacitivo diferencial [2].	19
2.26	Screenshot do ambiente de desenvolvimento da ferramenta Unity3D	20
3.1	Esquema de comunicação utilizado.	21
3.2	Tela de Login exibida pelo dispositivo com o jogo detectado 3.2a ou não 3.2b	22
3.3	Classes da aplicação nativa.	23
3.4	Classes da aplicação mobile.	23
3.5	Diagrama de sequência de comunicação.	24

3.6	Dois jogadores se enfrentando dentro do jogo.	24
3.7	Personificação do jogador	25
3.8	Telas relativas à escolha de personagem e calibragem dentro do jogo	26
3.9	Calibragem do personagem da esquerda	26
3.10	Autômato da colisão válida.	27
3.11	<i>Heads Up Display</i> - HUD - do jogo.	28
3.12	Jogo em execução.	28
4.1	Movimento de Yaw	29
4.2	Movimento de Pitch	30
4.3	Movimento de Roll	30
4.4	Movimentos dos dois braços.	31
4.5	Movimento de inclinação do tronco.	31
4.6	Movimentação do braço esquerdo.	32
4.7	Movimentação do braço direito.	32
4.8	Movimento dos joelhos.	33
4.9	Movimento das pernas.	33
4.10	Movimentos frequentemente utilizados no jogo.	34
4.11	“Achei o sistema fácil de usar.”	35
4.12	“O sincronismo - tempo de resposta - do jogo foi aceitável.”	36
4.13	“Me diverti jogando.”	36
4.14	“Acho a utilização do celular em conjunto com o <i>Kinect</i> uma alternativa viável, em termos de jogabilidade.”	37
4.15	“Não acho que precisaria de ajuda de um técnico para usar o sistema.” . . .	37
4.16	“Imagino que a maioria das pessoas aprenderia a utilizar o sistema rapidamente.”	38
4.17	“Compraria um jogo que apresentasse essa funcionalidade.”	39
4.18	“Este sistema apresenta vantagens em relação ao modelo convencional de jogos de luta.”	39
4.19	“Sou capaz de imaginar outras formas de utilizar o sistema de integração.”	40
4.20	“Se tivesse de comprar um aparelho específico para a simulação dos movimentos da espada, seria um incômodo.”	40

Lista de Tabelas

4.1	Resumo dos resultados obtidos	35
-----	---	----

Capítulo 1

Introdução

A área de jogos eletrônicos tem sido responsável por alavancar diversas e importantes evoluções no campo da computação, tais como a computação gráfica, inteligência artificial, simulação de habilidades, processamento de informação, etc. Temos, nessa área, o conceito de imersão, definido por Brown e Carnis como o grau de envolvimento do jogador com o jogo [5] e dividido em três graus, com barreiras de transição bem definidas. São eles:

- Engajamento, no qual o jogador tem interesse no jogo e deseja continuar jogando. Para entrar nesse estágio o jogador deve encontrar interesse pelo jogo e as características que o compõe e deve encontrar um controle preciso e/ou de fácil utilização.
- Absorção, no qual o jogador tem menos consciência do mundo a sua volta e de si mesmo enquanto joga, as características do jogo, como histórias, visual, tarefas, etc. se combinam de tal forma que o jogo passa a afetar diretamente a emoção do jogador.
- Imersão Total, no qual o jogo é a única coisa que impacta os pensamentos e sentimentos do jogador. Ele esquece que está jogando e sente-se como se estivesse dentro do jogo. Esse estado é alcançado quando existe empatia para com os personagens e sua causa e quando a atmosfera do jogo se combina de forma a tudo chamar a atenção apenas para o jogo, tudo dentro do jogo é relevante e deve ter foco apropriado.

Visando alcançar o primeiro estágio - engajamento -, novas tecnologias de interação do usuário estão sendo criadas para explorar essa experiência do jogador, vide infográfico disposto na Figura 1.1, que demonstra a evolução dos controles e seus maiores expoentes.



Figura 1.1: Infográfico de evolução dos controles de console da PopChartLab [24].

Os melhores exemplos de tentativas bem-sucedidas nesse quesito são apresentados a seguir na Figura 1.2.



Figura 1.2: Controles modernos

- *WiiRemote Plus* (Figura 1.2a) : utilizado como controle do *Nintendo Wii*, conta com acelerômetro, giroscópio e sensores infra-vermelhos para capturar movimentos da mão, permitindo um rastreamento preciso da movimentação do pulso do jogador.
- *Sony Playstation Move* em conjunto com o *Sony Playstation Eye* (Figura 1.2b) : utilizam processamento de imagens e um marcador óptico colorido, permitindo o rastreamento da parte superior do corpo nos consoles *Sony Playstation 3* e *4*.
- *Microsoft Kinect* (Figura 1.2c) : controle do *Microsoft Xbox 360*, utiliza processamento de imagens e mapeamento infra-vermelho de profundidade, permitindo o rastreamento de todo o corpo do jogador, porém com um nível de detalhamento restrito à cabeça, tronco e membros.

Cada um dos três sistemas revolucionou de alguma forma o modo como as pessoas interagem com o ambiente do jogo, extrapolando o simples apertar de botões. Uma solução que integre as melhores características destes sistemas é tecnologicamente possível, porém com um aumento do custo para o usuário. Como, então, seria possível um sistema que integrasse essas soluções sem onerar o possível jogador? Uma solução seria utilizar aparelhos que possivelmente esse jogador já tem para fazer as vezes dos sensores. Pensando nessa linha sabemos que os aparelhos celulares e *smartphones* são bastante populares. De acordo com a *AG2 Publicis Modem* [12], em 2013, 26% da população brasileira e 56% da população estadunidense já possuía algum desses dispositivos, conforme mostrado na Figura 1.3. Os modelos lançados a partir da segunda metade da primeira década do século XXI passaram a incorporar diversos sensores, entre os quais estão o acelerômetro e o giroscópio.

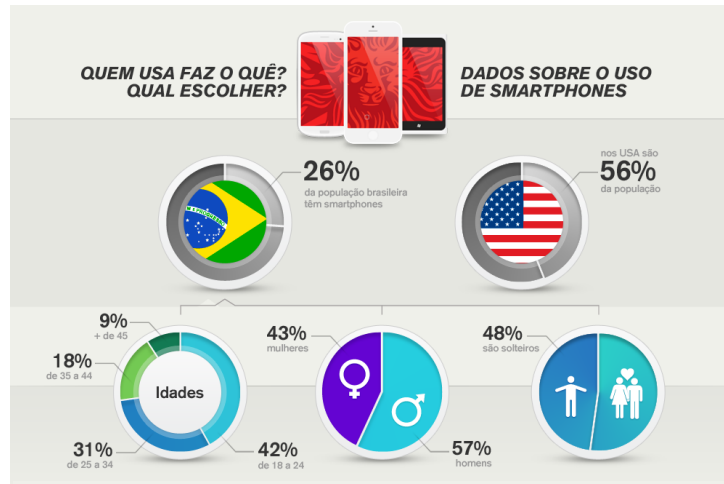


Figura 1.3: Infográfico disponibilizado pela AG2 Publicis Modem [12]

Este trabalho apresenta uma nova forma para desenvolvimento de jogos e sistemas que exijam o uso conjunto do giroscópio e do acelerômetro contidos em *smartphones* e do Kinect sem aumento significativo do custo para o usuário. Acredita-se na utilização do *Microsoft Kinect* em conjunto com um celular *smartphone*, como uma solução para a criação de sistemas que requeiram tanto o rastreamento preciso do corpo quanto do movimento da mão do usuário. Admitindo-se que um celular *smartphone* seja um item comum aos usuários de jogos eletrônicos, supõe-se que a utilização desta solução não gere um custo adicional para o usuário que já disponha de um aparelho *Kinect*.

Este trabalho, então, tem como objetivo principal o desenvolvimento de uma interface para a integração do dispositivo *Kinect* com sensores disponíveis em *smartphones* - especificamente, giroscópio e acelerômetro, e sua aplicação na criação de um jogo que exija o rastreamento preciso do corpo e dos movimentos do pulso do jogador. Os resultados obtidos são comparados com os sistemas de rastreamento existentes e o jogo será avaliado através de uma pesquisa de opinião.

O trabalho está assim dividido: o segundo capítulo apresenta um estudo sobre o estado-da-arte das técnicas de rastreamento corporal, de mãos e trabalhos que tiveram proposta semelhante. A solução proposta é apresentada com detalhes no capítulo três. O capítulo quatro descreve os resultados obtidos. Por fim, o capítulo cinco conclui o trabalho e apresenta sugestões de trabalhos futuros.

Capítulo 2

Referencial Teórico

Ferramentas de rastreamento de movimento corporal têm sido alvo de estudos dentro de várias áreas. Suas aplicações passam por segurança, telepresença, saúde, jogos, entre outras. Este capítulo aborda as principais técnicas e equipamentos encontrados na literatura e utilizados no desenvolvimento de ferramentas de rastreamento de movimentos corporais.

2.1 Trabalhos Correlatos

2.1.1 Rastreamento da Mão

Diversas aplicações exigem a correta identificação da posição, postura e movimentos das mãos. As principais técnicas podem ser divididas nas que usam visão computacional e nas que são baseadas em instrumentação eletrônica. As técnicas de visão envolvem a detecção e extração da imagem da mão através de técnicas de processamento de imagens e vídeos. As técnicas baseadas em instrumentação eletrônica utilizam-se de sensores estrategicamente posicionados na mão, tais como as luvas instrumentalizadas [13].

Robert Y. Wang. propõe o uso de duas câmeras RGB-D (*Red Green Blue Depth*) obtendo o rastreamento de ambas as mãos, mesmo de partes encobertas [43]. São posicionadas duas câmeras com angulação de 45 graus do seu eixo principal e 67 graus do plano onde se encontra o computador. Segundo os autores, essa configuração permite um espaço operacional de 34cm x 46cm x 24cm para a livre movimentação das mãos, proporcionando 6 graus de liberdade. A Figura 2.1 mostra a utilização do sistema. Os autores reportam uma alta precisão para movimentos específicos já treinados - 99% para um movimento de *pinch*, porém em um espaço de atuação bastante reduzido.

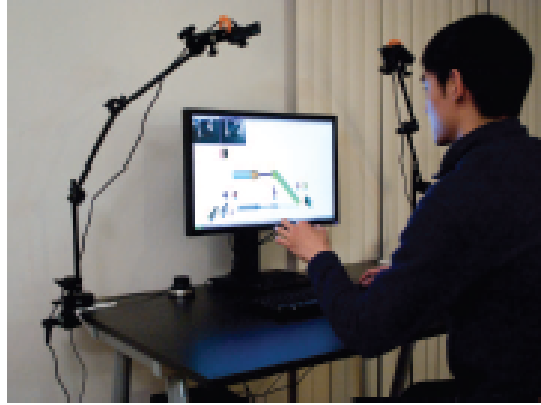


Figura 2.1: Rastreamento utilizando duas câmeras [43]

Em [36], Javier Romero propõe o uso de uma câmera comum para fazer o rastreamento da mão e do objeto manipulado por esta, como mostrado na Figura 2.2. Os resultados relatados apresentam uma baixa taxa de iteração e um reconhecimento insuficiente para a utilização em nosso projeto.

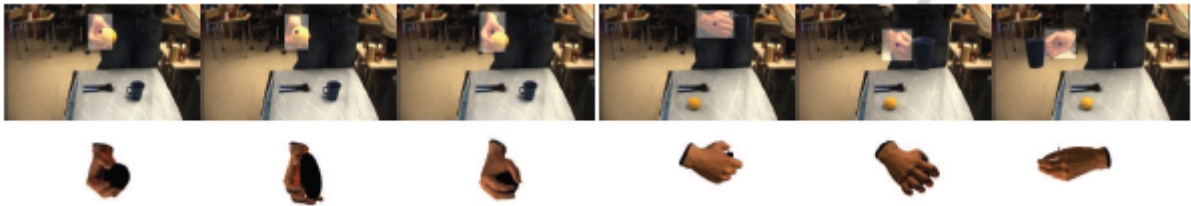


Figura 2.2: Rastreamento da mão com interação com objeto [36]

Oikonomidis *et al.* usam uma câmera RGB-D juntamente com uma placa de aceleração gráfica para fazer o rastreamento de articulações [30], como pode ser visto na Figura 2.3. A uma distância de meio metro tem-se um erro médio de 5mm, enquanto a 2.5m o erro é de 7.5mm. Para melhorar a taxa de captura, é utilizado o algoritmo PSO (*Particle Swarm Optimization*), baseado em um conjunto de possíveis imagens e escolhendo as mais adequadas utilizando o ponto mais próximo como critério.



Figura 2.3: Rastreamento da mão utilizando PSO [30]

Em um segundo trabalho, Oikonomidis evolui o sistema conseguindo rastrear a mão em conjunto com o objeto que segura, usando o método HOPE - *Hand-Object Pose Esti-*

tion [31]. Conforme apresentado na Figura 2.4: a palma da mão no modelo proposto é modelada como um cilindro elíptico e duas elipsóides, enquanto os dedos consistem em três cones e quatro esferas, exceto o polegar que consiste em 2 cones, uma elipsóide e três esferas. São utilizadas 25 esferas para o modelo de colisão das mãos, assim, pode-se checar a interação entre as mãos e outros objetos. Usa-se um sistema multicâmera com 8 câmeras sincronizadas e calibradas para evitar a falta de dados por sobreposição e o algoritmo PSO para encontrar a melhor proposta de modelo.

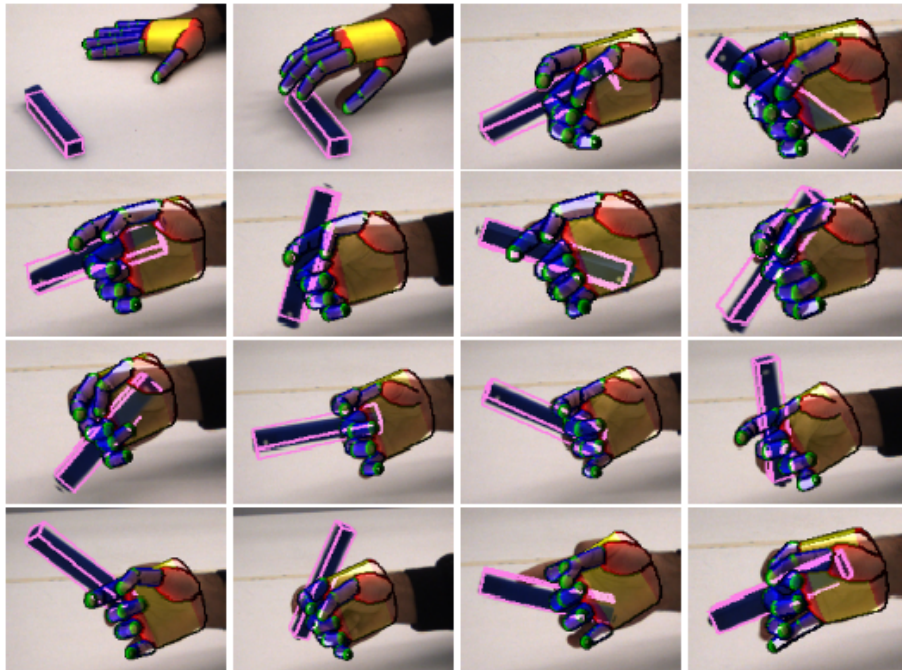


Figura 2.4: Rastreamento da mão com interação de objeto [31]

Lin *et al.* usam a posição de mão aberta, como na Figura 2.5, para reconhecimento e um algoritmo baseado em partículas e segmentações da mão para fazer rastreamento. Porém a solução não trabalha com partes obstruídas e nem com a mão reversa [26].



Figura 2.5: Rastreamento da mão proposta em [26]

Em outro trabalho, Tang *et al.* decompõem a posição da mão de acordo com cada dedo, montando um mapa de bits. O sistema usa a cor da pele para identificar a posição da mão no ambiente e uma região crescente para distingui-la [41]. A Figura 2.6 mostra a segmentação obtida.



Figura 2.6: Utilização do algoritmo proposto em [41]

2.1.2 Rastreamento corporal

Além da identificação e rastreamento das mãos, nossa solução faz necessário obter estimativas da postura corporal, de modo a permitir o rastreamento associado de mãos e corpo.

Rakate *et al.* utilizam transformação DWT (*Discrete Wavelet Transformation*) em 3 níveis para retirar informações de baixa frequência da imagem e o algoritmo *mean shift* para reconhecimento de objeto [34]. Dotado de uma câmera USB conectada a um sistema de CPU ARM, o sistema proposto foi capaz de processar em tempo real, a taxa de 25 *frames* por segundo. A Figura 2.7 mostra o sistema de rastreamento proposto. Onde a Figura 2.7a é a imagem do quadro anterior, a Figura 2.7b trata-se do quadro atual, a Figura 2.7c diferença entre quadros e a Figura 2.7d a região que delimita o corpo.

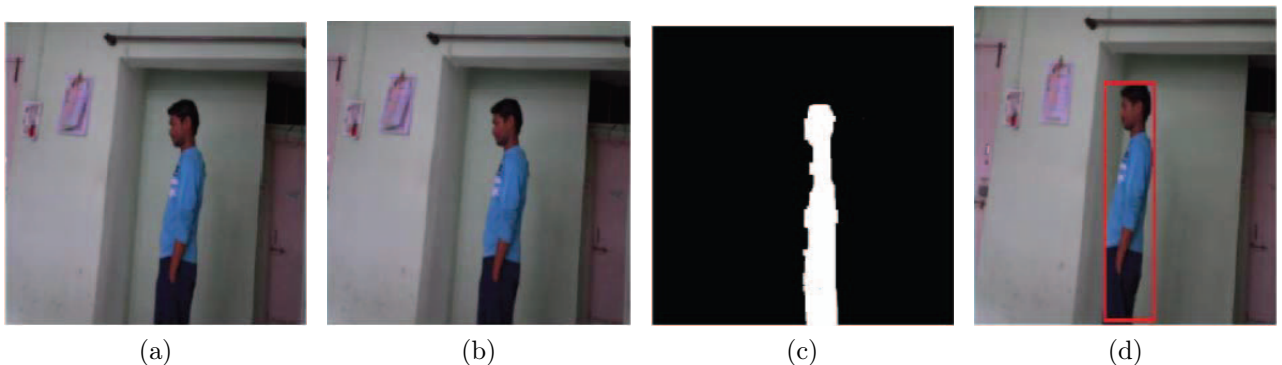


Figura 2.7: Rastreamento corporal utilizando [34].

Kant *et al.* propõem a utilização do algoritmo NCCA (Normalized Cross-Correlation of Average Human Histogram) para o rastreamento de múltiplos corpos humanos[22], usando câmera termal e sistema de cores HSV (*Hue Saturation Value*). Para detecção

do corpo é usado o possível intervalo de temperatura corporal. A Figura 2.8 mostra as passagens de quadros dentro do sistema. A necessidade de uma câmera termal é um impedimento, uma vez que a solução busca a simplicidade para o usuário final.

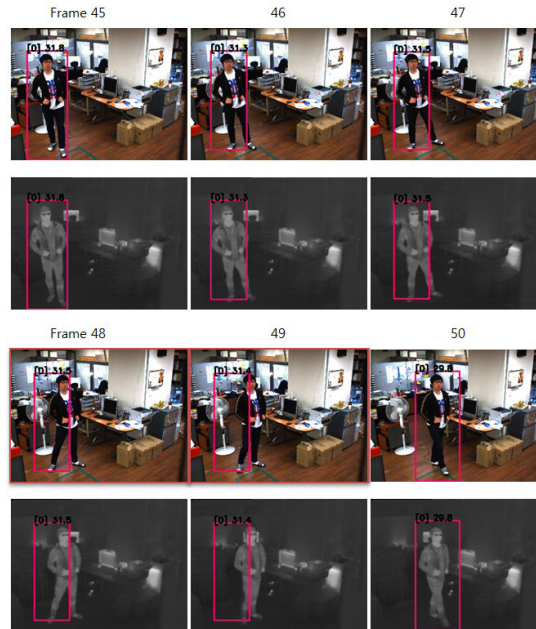


Figura 2.8: Rastreamento por temperatura corporal [22]

Lee *et al.* usam uma grade de sensores termais e de distância para executar o rastreamento do corpo inteiro dentro de uma sala[25], Figura 2.9. Não é viável como uma solução para o nosso problema, devido à aparelhagem sofisticada.

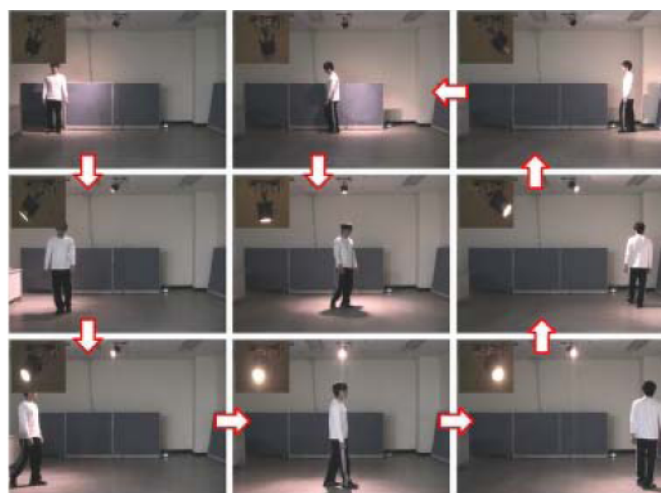


Figura 2.9: Localização do sujeito dentro de uma sala [25]

Ganapathi e Plagemann usam imagens de profundidade com uma câmera *Time-of-Flight - Swissranger SR4000* -, e uma GPU [15] para a captura de imagens e modelagem do corpo. Sua solução modela o corpo em 15 partes rígidas de acordo com um esqueleto com forma análoga a um grafo acíclico com relação pai-filho bem definida, sendo utilizado para modelagem probabilística do corpo. Determina-se, para a detecção, que o corpo humano é constituído de 5 partes: cabeça, mão esquerda, mão direita, pé esquerdo e pé direito. Usam-se os extremos e classificadores já treinados para o reconhecimento das partes. Dada uma correspondência entre um ponto na imagem e um ponto no modelo se utiliza inferência MAP (*Maximum a Posteriori*) para linearizar o problema de posição. Após a linearização do problema, é possível testar as hipóteses de posições equivalentes e obter a mais provável. A Figura 2.10 mostra um exemplo do modelo gerado de um participante dentro dos testes realizados.

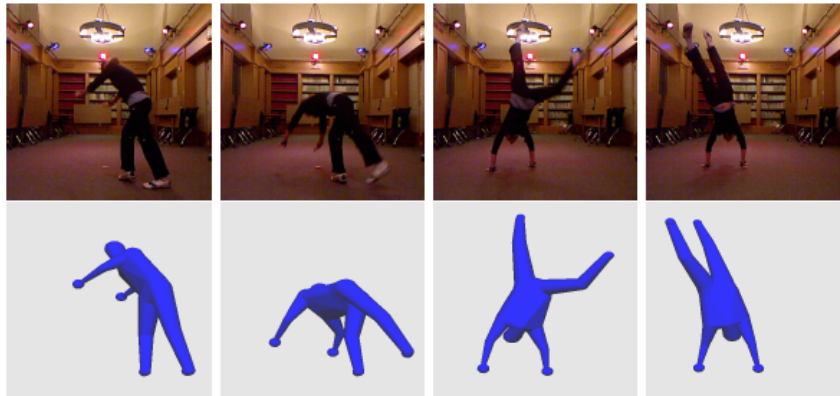


Figura 2.10: Modelo gerado pela movimentação de um participante em [15].

Em um trabalho posterior, ilustrado na figura 2.11, Ganapathi [16] usa novamente um grafo e inferência MAP para detecção de posição, adicionando a utilização do algoritmo ICP (*Iterative Closest Points*) em conjunto com uma forma do modelo de *Ray Casting*, e chamando o novo método de *Ray-Constrained ICP Model*. Para o modelo de transição, estabelece prerrogativas de movimento que um corpo humano não consegue fazer e assume um conjunto de possíveis estados futuros dada uma visualização passada.

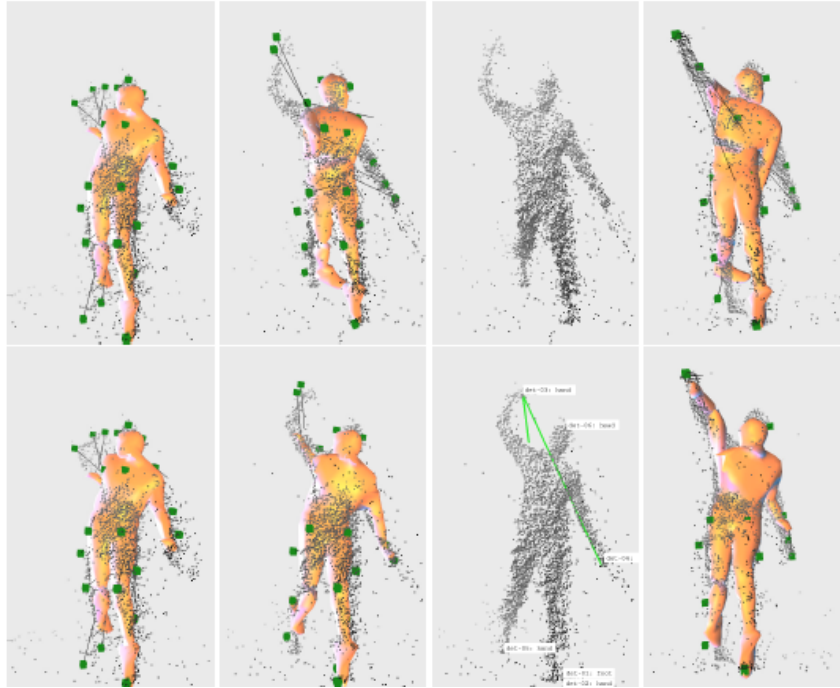


Figura 2.11: Captura de quadros utilizando o *Ray-Constrained ICP Model* [16]

2.1.3 Trabalhos Semelhantes

Os trabalhos citados nas seções anteriores utilizavam dos mais variados recursos para executar o rastreamento de mãos e corpo. Esta seção foca em soluções similares à proposta nesse trabalho.

Em 2010, Malfatt *et al.* apresentaram um *framework* de comunicação entre celulares e aplicações *desktop* [28]. A proposta, de nome *Blue Wave* e implementada em *Java*, é capaz de reconhecer até 7 dispositivos celulares simultaneamente uma vez que todos contenham uma máquina virtual *Java*. A Figura 2.12 mostra uma das utilizações do *framework*.



Figura 2.12: Sistema sendo utilizado em dois celulares para controle de um jogo de corrida *multiplayer* [28]

Mark Joselli e Eduardo Soluri, desenvolveram uma outra arquitetura baseada em *sockets* TCP/IP, onde há um maior número de informações obtidas a partir do dispositivo móvel, pois já trabalha com celulares dotados de *touchscreen* e acelerômetro. Restrito apenas a aparelhos com o sistema operacional iOS, a aplicação cliente é desenvolvida em Objective-C [21]. Na Figura 2.13 temos a imagem de uma aplicação utilizando a arquitetura para o controle de um jogo. O jogador pode utilizar a tela exibida no aparelho ou movimentar o celular para determinar a posição de uma nave nos eixos X e Y.



Figura 2.13: Framework sendo utilizado em um jogo funcional [21]

Em um trabalho de integração Vincent Tam e Ling-Shan Li, em 2012, propõem a utilização do *Kinect* no reconhecimento de gestos específicos, exemplo na Figura 2.14, ou comando de voz dentro de aplicações. Propõem também a geração de um *QR code* para que o aparelho celular possa se conectar e disparar comandos [40] ao *desktop*.

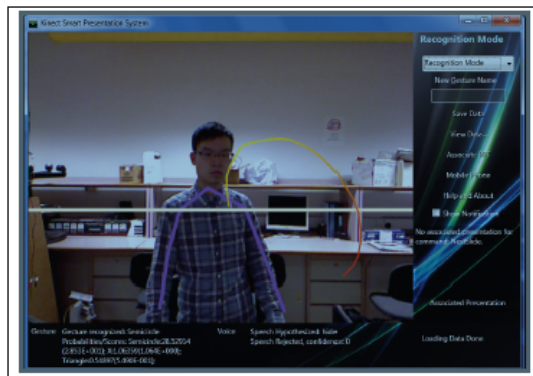


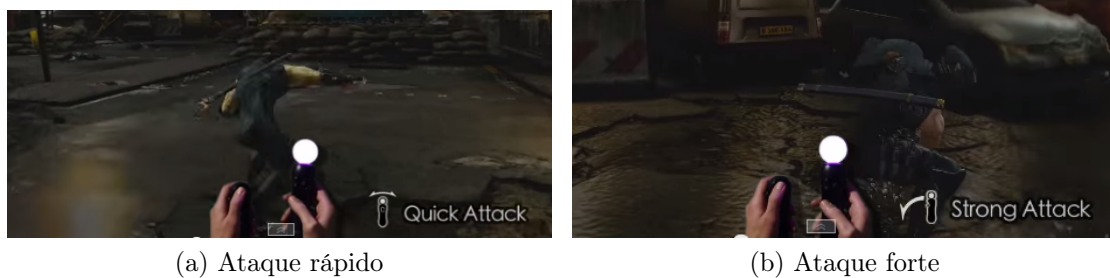
Figura 2.14: Reconhecimento de gestos em [40]

Na Universidade *Brown*, Michael Katzourin e sua equipe desenvolvem o *SwordPlay*, Figura 2.15, jogo que se utiliza 3 sensores com 6 graus de liberdade cada para trazer imersão ao jogador. Um é posicionado na cabeça a fim de controlar a visão do mundo e um em cada mão para o controle do jogador dentro do mundo do jogo. Para efetuar o deslocamento no mundo, utilizam-se controladores nas mãos do jogador[23].



Figura 2.15: Demonstração do *SwordPlay* em execução [23]

Uma aplicação comercial de utilização de sensores de movimento dentro de jogos é o *Ninja Gaiden 3* da *Team Ninja* e publicado pela *Tecmo Koei* do console *Playstation 3*. O jogo utiliza movimentos horizontais no *Playstation Move* para execução de ataques rápidos e movimentos verticais para execução de ataques fortes e mais lentos, ambos podem ser vistos na Figura 2.16.



(a) Ataque rápido

(b) Ataque forte

Figura 2.16: Jogo *Ninja Gaiden 3* da *Team Ninja* [20]

2.2 Kinect

O aparelho da *Microsoft* para rastreamento de movimento - *Kinect* - tem seu diferencial pela facilidade de fabricação e adaptabilidade. Rasteriza à taxas interativas e reconhece larga variação de formas corporais, tamanhos, vozes e faces. Utilizado inicialmente como interface para controle do console *Xbox 360*, hoje em dia tem sua utilização expandida em várias áreas, como fisioterapia, pesquisas acadêmicas, etc.

Nenhuma ferramenta de rastreamento, até então, conseguia executar em taxas satisfatórias ao mesmo tempo que lidava com uma grande variação de formas e tamanhos de corpos humanos.

Com o lançamento do console *Nintendo Wii* em 2006 utilizando um controle totalmente diferente para a época, o *Wii Remote*, a divisão responsável pelo *Xbox* da empresa *Microsoft* começa a pesquisa de um *input* de consoles que competisse com o mesmo. Utilizando a tecnologia desenvolvida pela empresa *PrimeSense*, formulou-se o projeto de um dispositivo que incluísse rastreamento de movimento, reconhecimento de voz, facial e de profundidade [18].

Em 4 de Novembro de 2010 pela empresa *Microsoft* para o seu console *XBox 360*, foi lançado o *Kinect* [8].

A câmera RGB-D captura imagens em profundidade e compara-as com uma base de dados sintéticos gerados previamente, permitindo ao software comparar movimentações já esperadas dentro da aplicação. O foco do *Kinect* é a captura de poses, então posições muito próximas são desnecessárias. Assim seus *subset* de poses tem tamanho de 100 mil unidades.

O corpo é dividido em partes e cada uma é reconhecida separadamente. O projeto de reconhecimento do *Kinect* teve como objetivos-chave a eficiência e a robustez [38]. Cada imagem é segmentada em várias partes prováveis de corpo, definidas para serem espacialmente localizadas perto de juntas interessantes ao reconhecimento do esqueleto humano. Re projetando as partes inferidas no espaço do mundo, pode-se gerar propostas com um bom grau de confiança para a localização 3D de cada junta do esqueleto humano. Cada segmentação do corpo é uma tarefa de classificação por *pixel*. É utilizado uma floresta de árvores de decisão para definir as partes do corpo, em conjunto com a posição.

2.3 Giroscópio

Giroscópio é um dispositivo de determinação ou manutenção de orientação utilizando o momento angular. Tem utilização ampla, servindo tanto para manutenção da orientação de aviões, proteção de equipamentos de computação, orientação de robôs e espaçonaves e para uso em dispositivos móveis, como celulares, controles de videogames, entre outros.

A empresa *Nintendo*, por exemplo, utiliza um giroscópio *InvenSense's IDG-600* no *Wii MotionPlus* para aumentar a precisão de seu *Wii Remote* [7].

Existem diversos tipo de giroscópios, entre os quais podemos citar:

- Girostato - sistema de corpos rígidos e ou partículas onde os centro de massa e inércia são independentes da configuração. Normalmente composto de um rotor suspenso por um suporte formado por dois círculos articulados.
- MEMS - Utiliza o efeito Coriolis para definir a direção e a mudança da mesma. Em geral é o giroscópio utilizado em celulares e outros dispositivos móveis por ser fácil de fabricar e de baixo custo [42].
- FOG - Utiliza o efeito de Sagnac para medir a rotação. Segundo ele, feixes de luz que se propagam em direções opostas em um círculo de raio R , percorrerão a mesma distância com a mesma velocidade, tão logo, se encontrarão no ponto de término ao mesmo tempo, porém se houver movimentação no círculo eles se encontrarão em fases diferentes e essa diferença será proporcional à rotação do mesmo [6].
- DTG - Baseado num rotor suspenso por uma junta universal com pontos de flexão. Traduz a velocidade angular utilizando o torque do gimbal e do rotor [11].
- London Momentum - Utiliza um super condutor giratório que gera um campo magnético, cujos eixos se alinham exatamente com o giro do eixo do rotor. Um magnetômetro determina a orientação do campo gerado e o interpola para definir o eixo de rotação.

Neste trabalho, utilizamos os giroscópios nos celulares para obter a informação de movimento do pulso do jogador, juntamente com o acelerômetro. Normalmente tratam-se de giroscópios do tipo MEMS, como já definido. A aquisição de informação deste tipo é feita da seguinte forma.

Um corpo de massa m se movimenta na direção \vec{v} e com velocidade angular de rotação $\vec{\Omega}$ aplicada, haverá a ação de uma força perpendicular \vec{F} à direção \vec{v} e à normal de rotação do corpo. A essa chamamos de força *Coriolis* e o deslocamento causado por essa é lido por sensores capacitivos.

A maioria dos giroscópios MEMS usa dois corpos que oscilam e movem-se constantemente em direções opostas. Quando uma velocidade angular é aplicada, a força *Coriolis* de cada corpo também age em direções opostas, que resultam em mudança de capacitância. A diferença em capacitância é proporcional à velocidade angular e pode ser convertida em voltagem para giroscópios analógicos ou valores LSB (Least Significant Bit) para digitais.

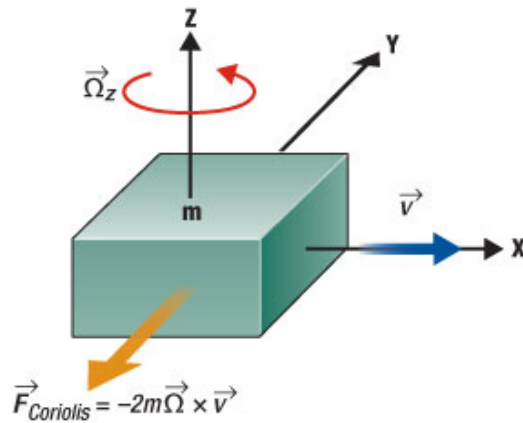


Figura 2.17: Força *Coriolis* para um corpo [42].

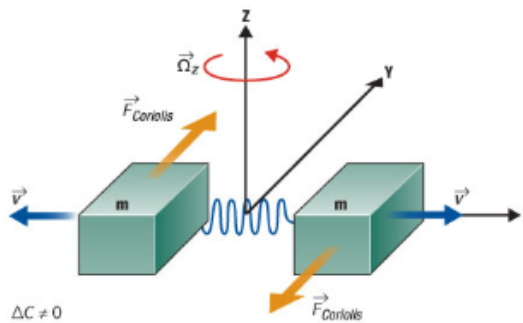


Figura 2.18: Esquema de funcionamento de um giroscópio MEMS [42]

2.4 Acelerômetro

Usado em conjunto com o giroscópio para determinar a rotação do pulso, o acelerômetro é um dispositivo eletromecânico de medição de força de aceleração. Sendo esta estática, como a força da gravidade ou dinâmica, causada por mover ou vibrar um acelerômetro, por exemplo [27][19].

Medindo a aceleração estática provocada pela gravidade, pode-se encontrar a rotação do dispositivo em relação ao solo. Medindo a aceleração dinâmica pode-se analisar a movimentação do mesmo. Entre os princípios físicos usados na fabricação de acelerômetros destacam-se alguns, citados a seguir.

- Capacitância (Figura 2.19) : Massa de prova produz diferença de capacitância proporcional à aceleração.



Figura 2.19: Exemplo de um acelerômetro capacitivo [17]

- Piezoelectricidade (Figura 2.20) : Constituído de dois elementos mais importantes:
 - Material Piezocerâmico.
 - Massa sísmica.

Quando o acelerômetro é sujeito à vibração, uma força é gerada no material piezoelétrico. O efeito piezoelétrico gera uma carga elétrica proporcional à força gerada. Como a massa sísmica tem massa constante, a carga gerada é proporcional à aceleração.



Figura 2.20: Exemplo de um acelerômetro piezoelétrico [9]

- Piezoresistividade (Figura 2.21) : Estrutura semelhante ao acelerômetro piezoelétrico, porém utiliza um material piezoresistivo, o qual, ao invés de gerar carga ao sofrer ação de força externa, gera alteração na própria resistência [37].



Figura 2.21: Exemplo de um acelerômetro piezoresistivo [29]

- Efeito Hall (Figura 2.22) : Conversão de movimento em sinais elétricos utilizando a mudança de campo magnético ao redor do acelerômetro.

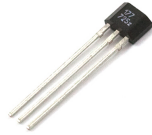


Figura 2.22: Exemplo de um acelerômetro utilizando o efeito Hall [33]

- Magnetorresistência (Figura 2.23) : Resistência do material muda na presença de campo magnético.



Figura 2.23: Exemplo de um acelerômetro Magnetoresistivo [1]

- Transferência de Calor (Figura 2.24) : Rastreamento de massas de calor capturadas durante a aceleração. Quatro termoresistores são dispostos a uma distância igual do centro de uma fonte de calor. Tendo aceleração zero, o calor estará disperso simetricamente, entretanto, dada qualquer variação na aceleração o calor se torna assimétrico graças à convecção do elemento de transferência de calor.

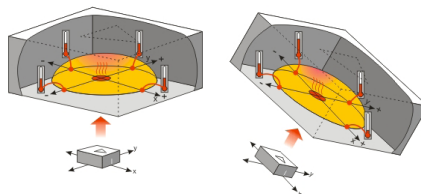


Figura 2.24: Exemplo de um acelerômetro usando transferência de calor [35]

Segundo Anthony Chu o acelerômetro mais comum utilizado em celulares hoje em dia é o capacitivo com eletrodos [9]. Dois eletrodos são fixos e paralelos e um livre localizado

entre os dois primeiros e paralelos. À medida que há movimentação do eletrodo livre, há também a mudança na capacitância dos eletrodos acima e abaixo. Essa mudança causa uma diferença entre a capacitância gerada entre o primeiro eletrodo e o livre e a gerada pelo segundo fixo e o livre. Tal diferença é proporcional à aceleração [3].

A capacitância entre dois eletrodos de mesma geometria e paralelos é:

$$C_0 = \epsilon\epsilon_0 \frac{A}{d} = \epsilon_A \frac{1}{d}$$

Onde $\epsilon_A = \epsilon\epsilon_0 A$, A é a área dos eletrodos, d é a distância entre eles e ϵ é permissividade do material que os separa. Adicionando um eletrodo móvel livre entre dois eletrodos temos duas novas capacitâncias C_1 e C_2 referentes a capacitância entre o primeiro eletrodo e o móvel e o segundo e o móvel, respectivamente. Dado que o móvel foi posto num ponto de distância d a partir dos dois outros. Utilizando as duas capacitâncias geradas pode-se obter a aceleração medida. Um esquema para melhor visualização pode ser observado na Figura 2.25.

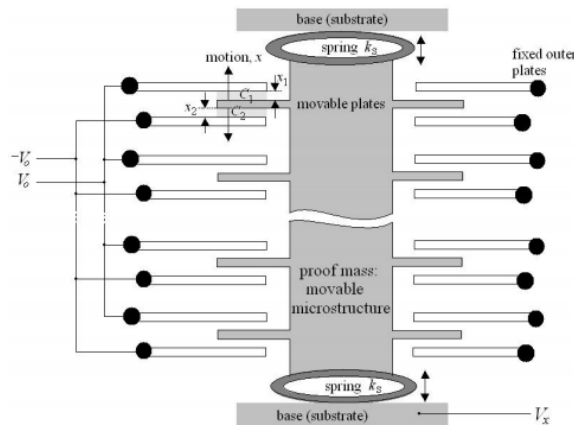


Figura 2.25: Esquema de um acelerômetro capacitivo diferencial [2]

2.5 Framework Unity

Unity3D trata-se de uma engine de desenvolvimento de jogos que contempla ferramentas para criação de jogos 2D e 3D e publicação multiplataforma. Sua primeira versão lançada em 2005 pela empresa *Unity Technologies*. Desenvolvida em C++ e C# sua função é tornar mais amigável a programação de jogos. A interface do jogo é produzida utilizando a abordagem WYSIWYG (*What You See Is What You Get*), como se pode observar pela Figura 2.26, enquanto a parte lógica pode ser desenvolvida utilizando as linguagens JavaScript, C# e Boo. Todo o processamento é realizado pela *Mono .Net*, implementação *open source* da plataforma *Microsoft .Net*. O Unity3D é de ampla utilização no mercado, principalmente no contexto *indie*, devido à facilidade de desenvolvimento e a quantidade de *assets* disponíveis produzidos pela comunidade. Foi considerada a alternativa mais viável na produção deste trabalho, pela questão da facilidade de implementação e seus

vastos recursos dentro da área de jogos.

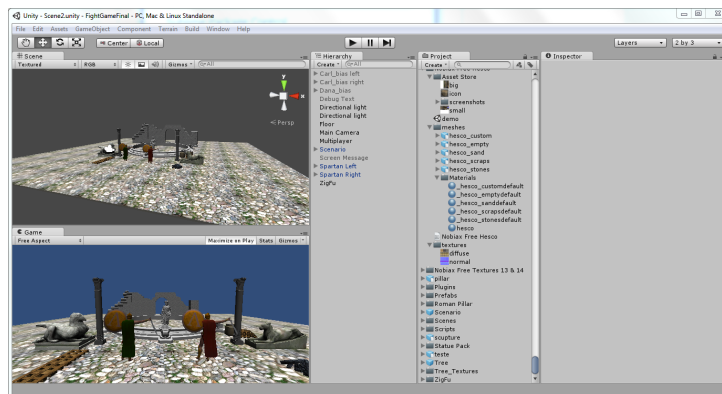


Figura 2.26: Screenshot do ambiente de desenvolvimento da ferramenta Unity3D

A *Unity* contém editores de terreno, de física, *shaders*, *scripts*, e controlador de versionamento e é capaz de publicar jogos para *Microsoft Windows*, *Mac OS*, *Xbox 360*, *Playstation 3*, *Nintendo Wii*, *IOS*, *Android* e *Web Player*. Exemplos de jogos já produzidos com ele são: *Knights of Pen and Paper* da *Behold Studios* [39], *Gone Home* da *Fullbright Company* [10], *Guns of Icarus Online* da *Muse Games* [14], entre outros. Sua interface extremamente amigável e seus *assets* produzidos pela comunidade facilitam o desenvolvimento de jogos.

Os dois *plugins* utilizados ao longo do trabalho são apresentados a seguir: um faz a captura de informações de todos os sensores de um dispositivo móvel com sistema operacional *Android*, enquanto o outro permite a utilização do *Microsoft Kinect* em conjunto com a plataforma *Unity*.

A *Unity* provê suporte nativo para a captura de informações básicas de alguns dos sensores de diversos dispositivos móveis. Com o *plugin GyroDroid* da *prefrontal cortex* é possível fazer a captura e processamento de dados de quaisquer sensores de aparelhos *Android* [32]. Foi priorizada a utilização de giroscópio e acelerômetro dentro da aplicação, sendo possível reproduzir a movimentação do pulso do jogador a partir da orientação obtida do celular.

O *plugin Zigfu*, em contrapartida, permite a utilização do *Kinect* em conjunto com a plataforma *Unity* [44]. Ele instala as versões mais recentes do *OpenNI*, um *framework* de propósito geral para a obtenção de dados de sensores 3D, *NITE*, uma biblioteca de rastreamento de esqueletos e de reconhecimento de gestos, e o *driver SensorKinect* para realizar a interface com o *Microsoft Kinect*. Apresenta também uma API para desenvolvimento e exemplos que facilitam o entendimento da ferramenta.

Capítulo 3

Implementação do Sistema

A fim de atingir os objetivos estabelecidos e conseguir uma experiência mais interativa para um jogo, propõe-se a utilização do equipamento *Kinect* em conjunto com um dispositivo móvel dotado de acelerômetro, giroscópio e capacidade de conexão via *WiFi*.

Para validar a solução, um jogo foi criado utilizando *Unity3D* e os *plugins* já citados. Este consiste em uma arena onde dois jogadores controlam dois gladiadores com o objetivo de derrotar um ao outro em uma batalha com espadas e escudos. A movimentação corporal desses é feita de modo a copiar os movimentos dos jogadores da maneira mais real possível, utilizando o *Kinect* por meio do *plugin ZigFu*. A movimentação da arma de cada jogador é calculada de acordo com os dados fornecidos pelos sensores dos celulares, utilizando o *plugin GyroDroid*.

Para isso, foram criadas duas aplicações. A aplicação nativa é executada em um computador *desktop* ou *laptop* com os sistemas operacionais Windows, MacOS ou Linux. A aplicação móvel é executada em um dispositivo móvel dotado de acelerômetro e giroscópio com o sistema operacional Android. A primeira aplicação é responsável por calcular a movimentação dos personagens, a colisão das espadas com os oponentes e apresentar a batalha em si aos jogadores entre outras, enquanto a segunda é responsável por calcular a orientação do dispositivo, conectar-se com a aplicação nativa via *WiFi* e enviar a orientação e outros dados. A Figura 3.1 mostra um esquema de como é a comunicação.

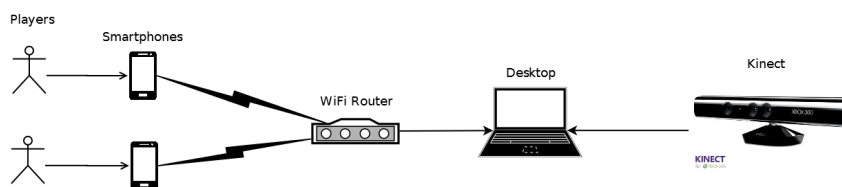


Figura 3.1: Esquema de comunicação utilizado.

3.1 Captura do movimento do celular

Para a implementação da comunicação dos celulares com a aplicação nativa foi utilizada uma arquitetura de computação distribuída, na qual os celulares servem de clientes

para o servidor, no caso, a aplicação nativa. Deste modo, os clientes enviam por RPC (*remote procedure call*) as informações de movimentação e aceleração dos celulares. Porém, a aplicação nativa assume o papel de cliente na ocasião em que ela detecta uma colisão da espada com outros elementos dentro do jogo, e envia uma requisição para que os celulares vibrem e reproduzam sons. Toda troca de mensagens é feita em uma rede sem fio, onde os celulares se conectam usando o endereço IP e porta disponibilizados pela aplicação nativa, inseridos na tela da Figura 3.2.

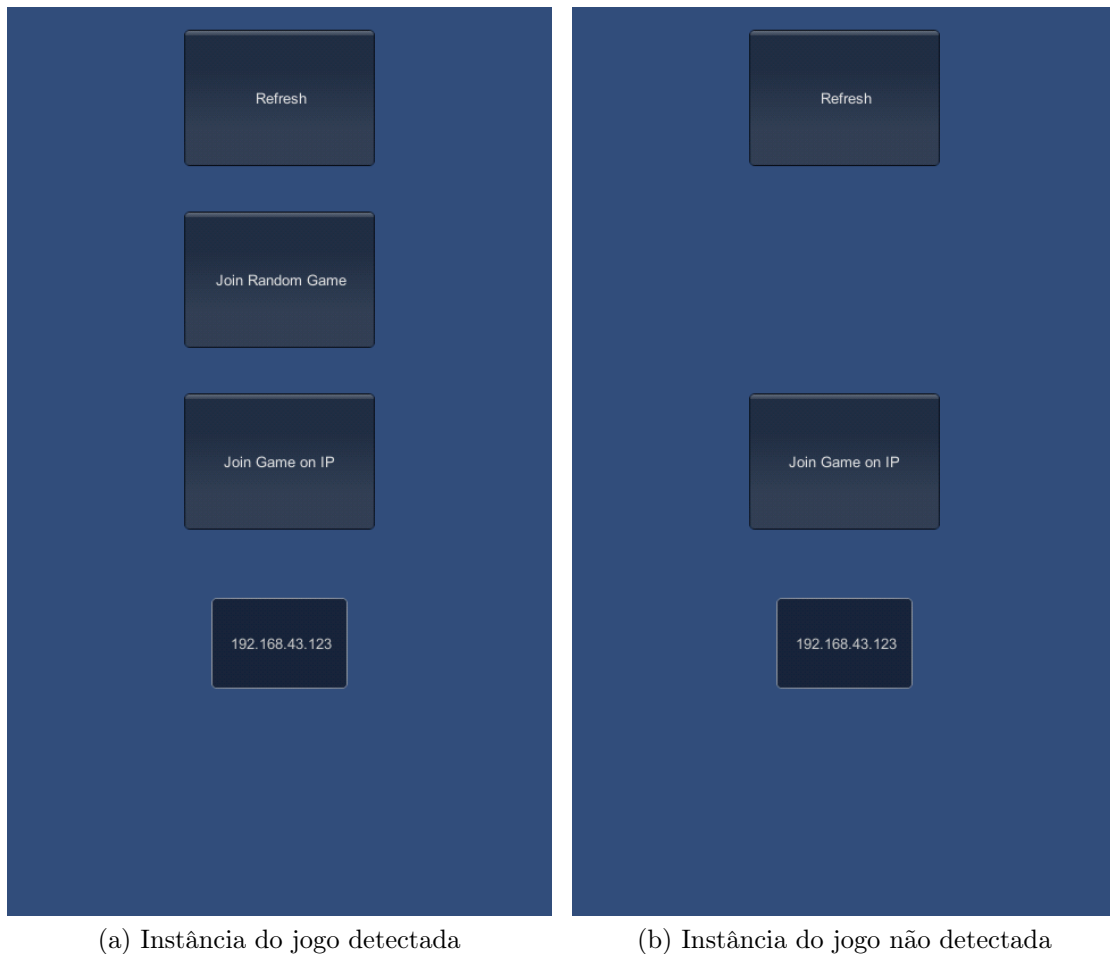


Figura 3.2: Tela de Login exibida pelo dispositivo com o jogo detectado 3.2a ou não 3.2b

A Figura 3.3 apresenta o diagrama das classes utilizadas na aplicação nativa, onde *MultiplayerMenu* é responsável pelo gerenciamento de conexões, *GameScript* trata-se da classe que registra cada jogador, *SendMobileData* faz a interface de RPC e tradução das rotações dos celulares para as espadas e, por fim, *SwordCollision* detecta as colisões das espadas dentro do jogo e envia comandos de vibração e de reprodução de sons para os celulares.

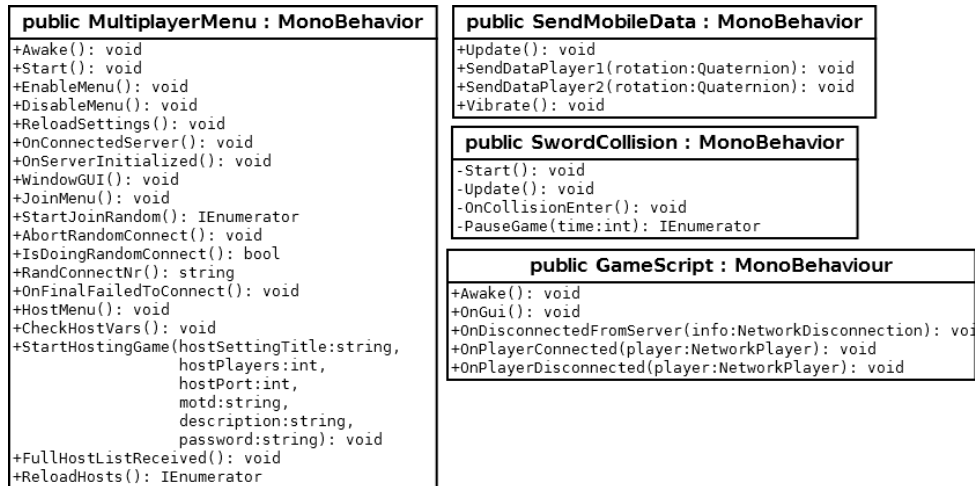


Figura 3.3: Classes da aplicação nativa.

As classes relacionadas à aplicação móvel são representadas na Figura 3.4: *SendMobileData* faz a interface de RPC, envia a rotação para a aplicação nativa e recebe o comando de vibração e reprodução de sons, *PlayerChoice* define a espada que cada personagem irá controlar e *Platform* define a plataforma do dispositivo.

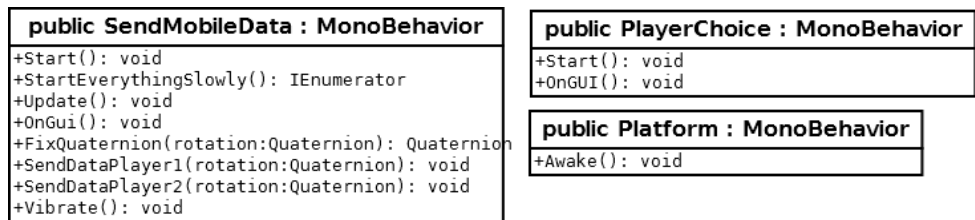


Figura 3.4: Classes da aplicação mobile.

Durante o jogo, infere-se uma orientação inicial às espadas até que os jogadores pressionem o botão para calibrar de cada aplicação móvel. Uma vez que o celular se movimenta, o evento é notado pela aplicação móvel e a rotação é capturada pelo acelerômetro e o giroscópio. A cada novo quadro (*frame*) da aplicação móvel, é feita uma chamada RPC. A informação da orientação obtida é enviada para aplicação nativa que atualiza a orientação da espada de acordo. A Figura 3.5 dá uma visão geral de como é feita a comunicação.

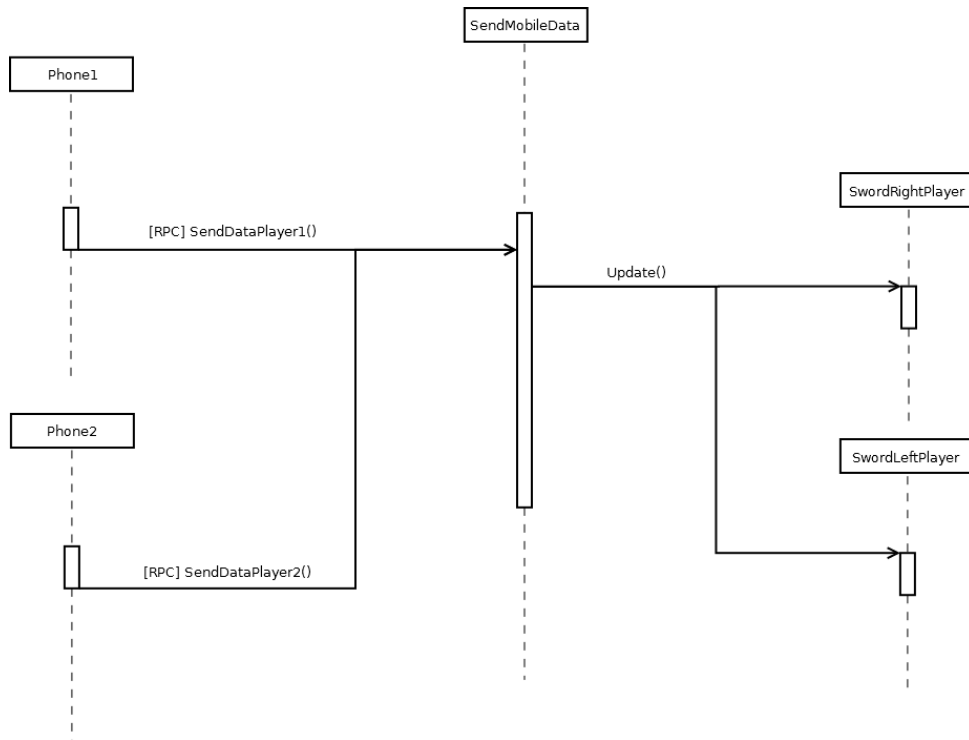


Figura 3.5: Diagrama de sequência de comunicação.

3.2 Captura do movimento corporal

Cada personagem do jogo se movimenta de forma a copiar os movimentos dos jogadores refletidos na direção ortogonal. Assim, quando ambos os jogadores estão voltados para a tela do jogo, os personagens controlados apresentam-se um de frente ao outro, visualizados lateralmente. Deste modo, pode-se ter uma experiência de combate real sem que haja dificuldades de visualização da tela do jogo ou que os jogadores venham a colidir um com o outro fora do jogo. A Figura 3.6 exemplifica o posicionamento.



Figura 3.6: Dois jogadores se enfrentando dentro do jogo.

O *plugin ZigFu* utiliza o *Kinect* para capturar as posições e rotações das diversas juntas do corpo de cada jogador. As juntas que podem ser capturadas: cabeça, pescoço, torso, cintura, clavícula esquerda, ombro esquerdo, cotovelo esquerdo, pulso esquerdo, mão esquerda, ponta da mão esquerda, clavícula direita, ombro direito, cotovelo direito, pulso direito, mão direita, ponta da mão direita, quadril esquerdo, joelho esquerdo, tornozelo esquerdo, pé esquerdo, quadril direito, joelho direito, tornozelo direito e pé direito.

Cada junta capturada é atualizada a cada novo quadro da aplicação nativa com os novos valores de posição e rotação correspondentes. Com a exceção da junta do pulso direito, que segue a orientação recebida da aplicação móvel.

3.3 O Jogo

Cada jogador é representado no jogo como um gladiador, apresentado na Figura 3.7, cada um dotado de uma espada e um escudo em meio a uma batalha um contra um.

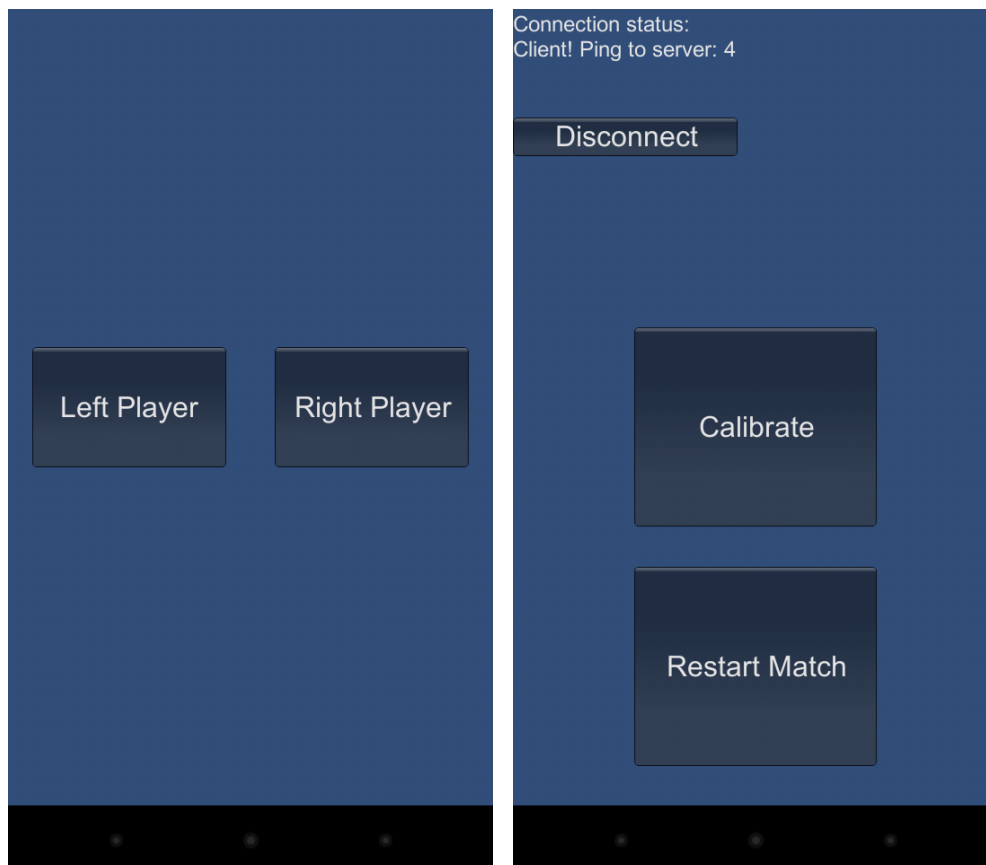


Figura 3.7: Personificação do jogador

É necessário que os jogadores tenham a aplicação móvel instalada em seu *smartphone*. Uma vez instalada, ela apresenta uma tela que habilita a conexão utilizando tanto o endereço IP ou procurando por um jogo aberto dentro da própria rede.

Ao conectar-se, o jogador deve escolher controlar o personagem posicionado à esquerda ou à direita, conforme pode ser visto na Figura 3.8a. O último passo se resume a calibrar o celular na tela da figura 3.8b, pressionando o botão calibrar estando de frente para a

tela da aplicação principal e em uma posição com a tela do celular virada de frente para ele em posição vertical - como na Figura 3.9, dando início à partida.



(a) Escolha de personagem.

(b) Tela para a calibragem.

Figura 3.8: Telas relativas à escolha de personagem e calibragem dentro do jogo



Figura 3.9: Calibragem do personagem da esquerda

O objetivo de cada jogador é causar dano ao personagem do adversário utilizando a espada. Uma vez que a espada colide com o adversário, o atacante deve retirar a espada de qualquer posição de colisão com o adversário e só depois de fazê-lo poderá causar dano novamente, assim evita-se que o jogador repouse sua espada dentro do adversário. Não há colisão em situações consideradas inválidas, tais como contra o próprio jogador ou encostando no chão. Quando há uma colisão válida, pode haver dois resultados, como descrito no autômato da Figura 3.10: fim de jogo, quando o membro onde a espada encostou é cortado e há a derrota de quem sofreu o dano ou uma diminuição da “vida” do personagem até que o dano causado seja suficiente para causar sua derrota.

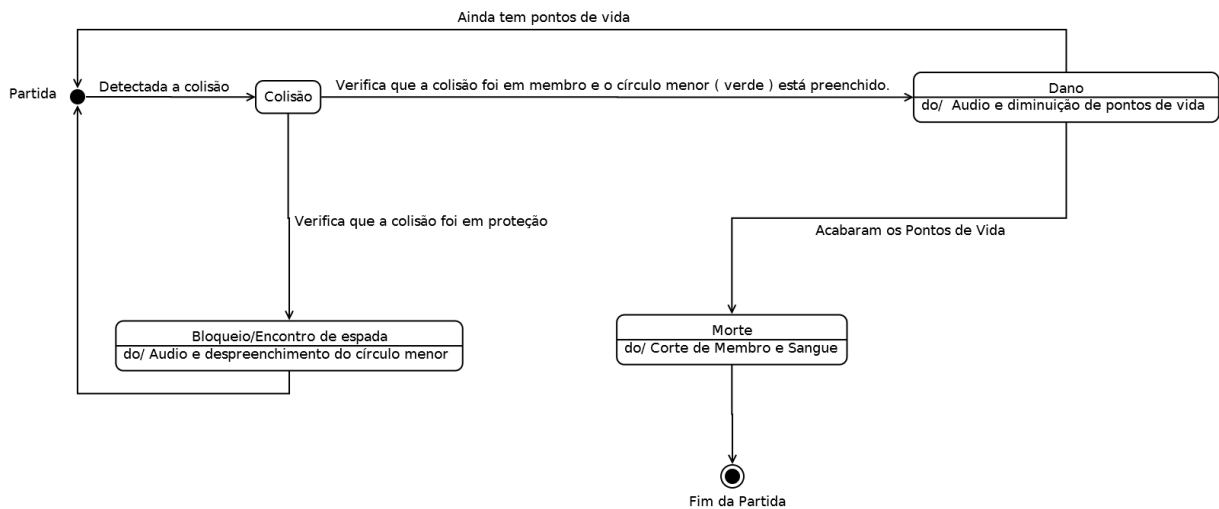


Figura 3.10: Autômato da colisão válida.

É estabelecido no jogo que existem 4 lugares onde o corte da espada pode causar dano, são eles: perna esquerda, perna direita, tronco e cabeça. Para melhorar a jogabilidade foi criado um sistema de “dano suportado” para cada personagem, em cada partida. Definiu-se que um número de 3 cortes em qualquer lugar válido exceto o tronco seria o suficiente para derrotar o adversário e se dê a animação de morte, exceto no tronco, onde o número aumenta para 7.

Para a visualização tanto do dano recebido pelo personagem, quanto do tempo necessário para que o mesmo possa atacar novamente uma vez feito um ataque, fazem parte do HUD (*Heads Up Display*) do jogo as duas esferas exibidas na Figura 3.11. A esfera maior (vermelha) representa a “vida” do personagem e perde preenchimento à medida que ele sofre dano. O círculo menor (verde), por sua vez, representa a possibilidade de atacar novamente quando estiver preenchido e a impossibilidade de gerar dano enquanto vazio. Para o círculo verde voltar a ficar preenchido, o jogador deve retornar a sua espada para suficientemente longe do oponente após a colisão da mesma com um alvo válido. Deste modo evita-se que um jogador cause dano ao oponente mantendo sua espada dentro deste ou fazendo apenas movimentos mais curtos do que o desejado para que haja uma boa experiência de jogo.

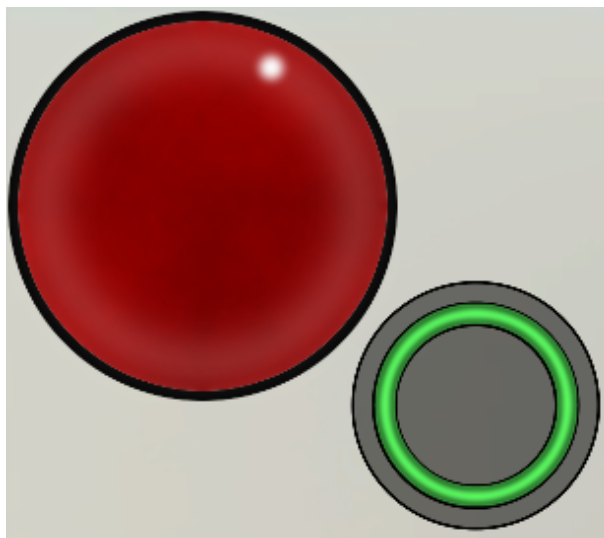


Figura 3.11: *Heads Up Display* - HUD - do jogo.

O jogo em execução com dois jogadores é apresentado na Figura 3.12, onde se pode observar dois jogadores dotados de celulares distintos jogando em frente a um *Kinect* ligado a um *MacBook*. Um vídeo hospedado na aplicação Youtube apresenta o sistema em ação, segue o link referente [Demonstração de TG K++ \[4\]](#).



Figura 3.12: Jogo em execução.

-

Capítulo 4

Resultados Obtidos

Os resultados da aplicação da metodologia proposta foram centrados em pesquisas e avaliação de dados do trabalho. O sistema nativo foi testado em um computador *laptop* com um processador Intel(R) Core(TM) i5-2450M @ 2.50GHz, 6.00GB de memória RAM, uma placa de aceleração gráfica NVidia GeForce GT 540M e sistema operacional Windows 7 Home Premium 64 bits. Já os testes do sistema móvel foram realizados nos *smartphones* G-I9500, Galaxy SII, Galaxy SIII, Galaxy SII Mini, GT-I8190L, GT-I9070, Moto X, Nexus 4 e Nexus 5, todos com o sistema operacional Android.

4.1 Resultados do rastreamento de pulso

São apresentados a seguir seqüências de imagens que mostram a posição real do pulso do jogador e seu correspondente manuseio da espada pelo personagem. Os resultados do rastreamento de pulso podem ser observados nas figuras a seguir.

A Figura 4.1 mostra o espelhamento da espada em relação a movimentação do celular em um movimento de rotação *Yaw*.

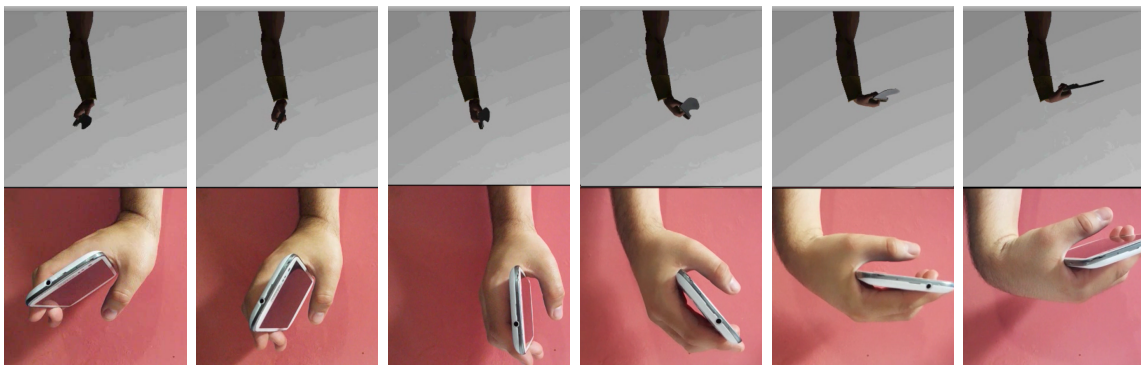


Figura 4.1: Movimento de Yaw

A Figura 4.2 mostra a comparação entre o movimento de rotação *pitch* do celular e o executado pela espada dentro do jogo.

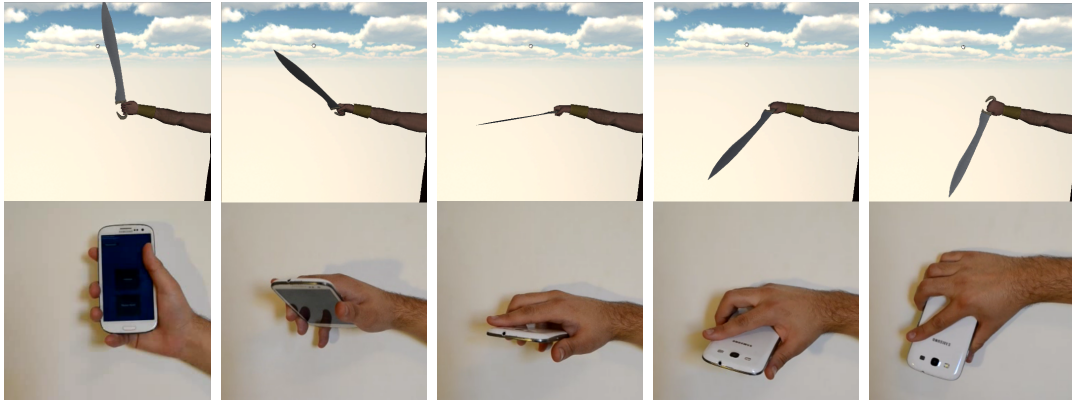


Figura 4.2: Movimento de Pitch

As rotações *Roll* do celular e espada são apresentadas para comparação na Figura 4.3.

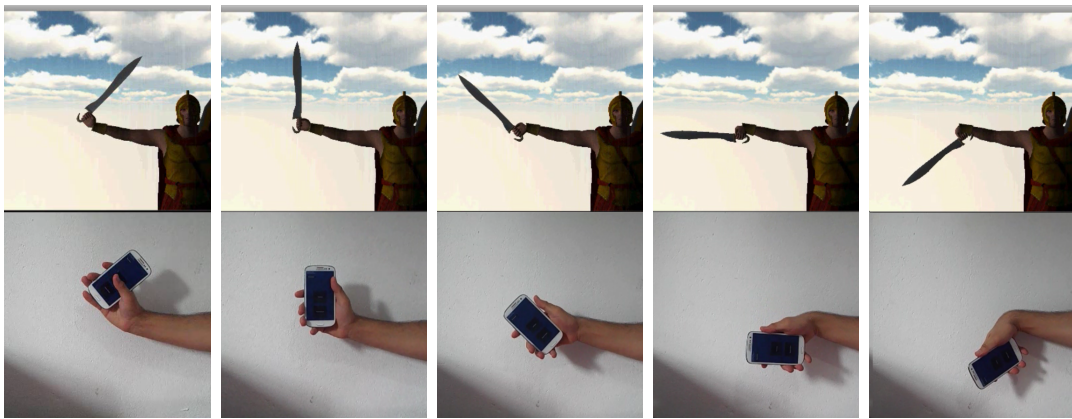


Figura 4.3: Movimento de Roll

Percebe-se que o uso dos sensores incorporados no aparelho celular permite um rastreamento preciso dos possíveis movimentos do pulso e os sintetiza corretamente no personagem do jogo.

4.2 Resultados do rastreamento corporal

As próximas figuras contêm imagens do jogador, da detecção feita pelo *Kinect* e do resultado final do jogo (observar que as imagens são capturadas invertidas pelo *Kinect*, invertendo a direita pela esquerda apenas nas imagens, não impactando no jogo). A Figura 4.4 mostra a movimentação dos dois braços em conjunto.



Figura 4.4: Movimentos dos dois braços.

A Figura 4.5 contém imagens para a comparação de movimentos com o corpo inclinado.

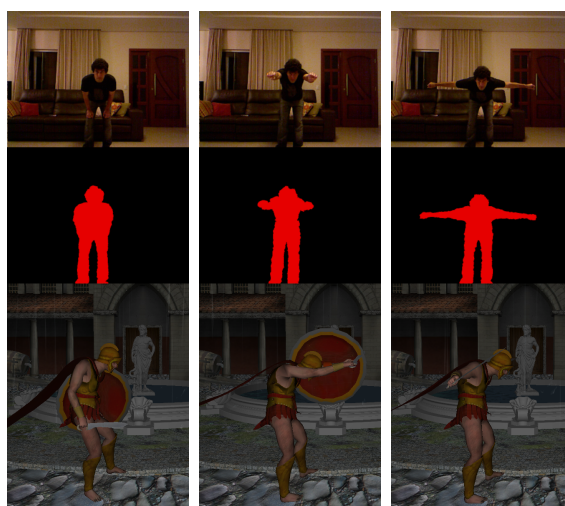


Figura 4.5: Movimento de inclinação do tronco.

A movimentação apenas do braço esquerdo, que segura o escudo, pode ser observada na Figura 4.6.

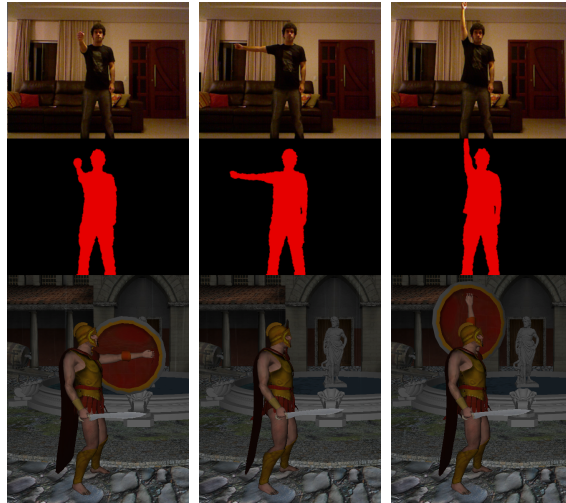


Figura 4.6: Movimentação do braço esquerdo.

A Figura 4.7 contém as imagens da movimentação do braço direito, que segura a espada.

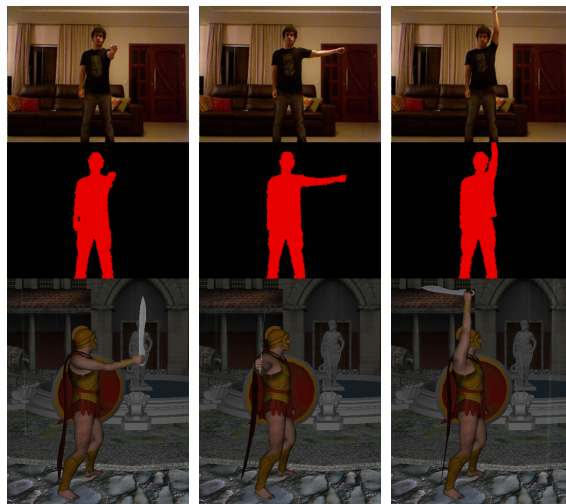


Figura 4.7: Movimentação do braço direito.

A movimentação de ambos os joelhos pode ser observada na Figura 4.8.



Figura 4.8: Movimento dos joelhos.

As pernas têm movimentos espelhados nas imagens contidas na Figura 4.9.



Figura 4.9: Movimento das pernas.

Por fim, os movimentos mais utilizados durante o jogo são apresentados na Figura 4.10, onde o escudo está em posição de defesa e a espada em posição de ataque.

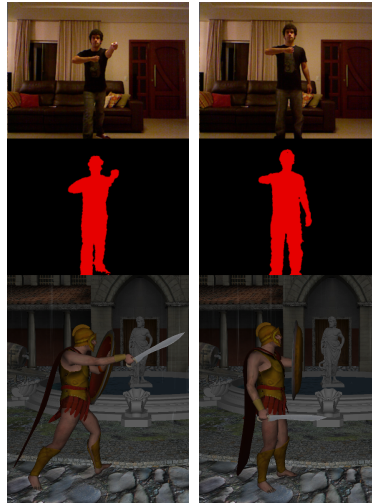


Figura 4.10: Movimentos frequentemente utilizados no jogo.

Percebe-se que o *Kinect* rastreia com sucesso os movimentos do corpo do jogador, permitindo que o sistema proposto sintetize adequadamente o personagem do jogo.

O uso conjunto do *Kinect* e sensores proposto neste trabalho permite o rastreamento do corpo e um detalhamento preciso dos movimentos do pulso, fundamentais para a simulação de uma luta com espadas.

4.3 Pesquisa de opinião

Com o objetivo de verificar a validade da proposta aplicada ao jogo uma pesquisa de opinião foi elaborada. Para tanto, solicitou-se que os participantes avaliassem seu grau de concordância com 10 (dez) afirmativas sobre o jogo desenvolvido nesse trabalho. A avaliação é feita atribuindo uma nota de 0 a 5 a cada afirmativa, onde 0 significa “total desacordo” e 5 significa “concordo completamente”. As 10 afirmativas são as seguintes:

1. “Achei o sistema fácil de usar.”
2. “O sincronismo - tempo de resposta - do jogo foi aceitável.”
3. “Me diverti jogando.”
4. “Acho a utilização do celular em conjunto com o *Kinect* uma alternativa viável, em termos de jogabilidade.”
5. “Não acho que precisaria de ajuda de um técnico para usar o sistema.”
6. “Imagino que a maioria das pessoas aprenderia a utilizar o sistema rapidamente.”
7. “Compraria um jogo que apresentasse essa funcionalidade.”
8. “Este sistema apresenta vantagens em relação ao modelo convencional de jogos de luta.”
9. “Sou capaz de imaginar outras formas de utilizar o sistema de integração.”

10. “Se tivesse de comprar um aparelho específico para a simulação dos movimentos da espada, seria um incômodo.”

Foi oferecido, também, um espaço no qual os participantes poderiam escrever comentários, críticas e/ou sugestões ao sistema.

A pesquisa foi realizada entre os dias 1 e 5 de Maio de 2014 no centro acadêmico do curso de Ciência da Computação da Universidade de Brasília e contou com 27 participantes voluntários entre 16 a 30 anos, utilizando celulares com a solução pré-instalada. A tabela 4.1 apresenta a quantidade de cada nota recebida para cada afirmação ao final da pesquisa. Pode-se perceber um número relativamente baixo de notas abaixo de 3, quando comparado com as notas superiores, indicando um bom nível de aceitação entre os participantes.

Notas:	0	1	2	3	4	5
Questão 1	0	0	0	7	10	10
Questão 2	0	0	1	9	14	3
Questão 3	0	0	0	2	7	18
Questão 4	0	0	1	2	8	16
Questão 5	2	0	1	3	8	13
Questão 6	0	0	0	6	8	13
Questão 7	1	1	1	6	8	10
Questão 8	0	0	1	9	10	7
Questão 9	3	2	2	6	7	7
Questão 10	0	4	2	8	5	8

Tabela 4.1: Resumo dos resultados obtidos

Na Figura 4.11 o gráfico apresenta a porcentagem das notas atribuídas à primeira questão.

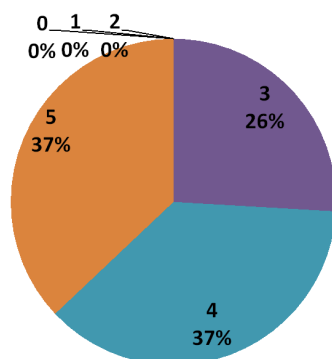


Figura 4.11: “Achei o sistema fácil de usar.”

Observa-se que a menor nota dada à primeira afirmativa foi 3, sendo a escolha de 26% de nossos participantes, enquanto 74% deu nota 5 ou 4. Este resultado indica que o sistema proposto foi considerado fácil de ser usado, mesmo por usuários leigos em computação. De fato, o sistema requer poucos procedimentos de ajustes e calibrações, e a mecânica do jogo é bastante intuitiva.

A Figura 4.12 contém o gráfico das porcentagens das notas acerca da questão 2 - “O sincronismo – tempo de resposta – do jogo foi aceitável.”

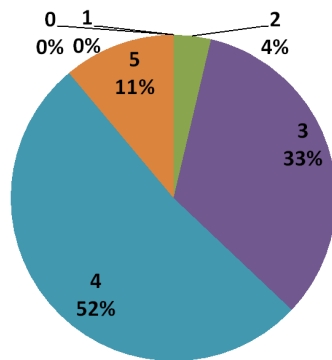


Figura 4.12: “O sincronismo - tempo de resposta - do jogo foi aceitável.”

Esta teve apenas 11% de aprovação total, 52% de nota 4, seguido de 37% de notas 2 e 3. Não havendo nenhum participante que avaliasse como 0 ou 1. Entretanto, a maior parte das críticas escritas referia-se ao sincronismo como uma das partes a melhorar do jogo.

As porcentagem de notas da questão 3 - “Me diverti jogando.” - são apresentadas na Figura 4.13.

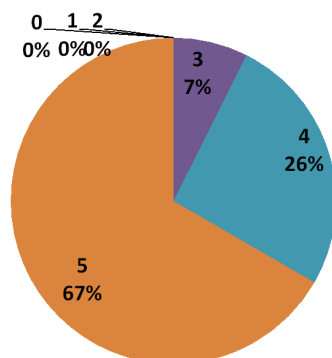


Figura 4.13: “Me diverti jogando.”

A pesquisa mostra 93% de notas entre 5 e 4, restando aos últimos 7% a nota 3. Isto indica que a solução proposta se apresenta como adequada para o jogo proposto, divertindo os participantes. Cabe ressaltar que a ideia de controlar um jogo com base em movimentos corporais reais, não é adequada em vários tipos de jogos, pois a movimentação requerida pode se tornar fatigante e incômoda, principalmente em jogos de longa duração.

A Figura 4.14 mostra as porcentagens das notas para a questão 4.

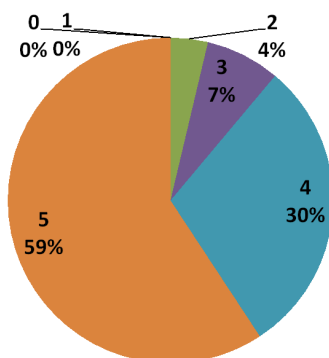


Figura 4.14: “Acho a utilização do celular em conjunto com o *Kinect* uma alternativa viável, em termos de jogabilidade.”

Uma das preocupações principais da pesquisa é sobre a opinião dos usuários finais sobre a utilização do celular em conjunto com o *Kinect*. A pesquisa mostra que 59% dos participantes concordam completamente que é viável, 30% deram nota 4, 11% deram notas entre 2 e 3. Indicando, dentro do escopo da pesquisa, a existência de uma aceitação da alternativa de jogo híbrido.

A indicação das porcentagens das notas da questão 5 é feita pela Figura 4.15.

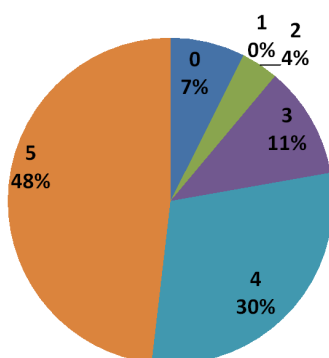


Figura 4.15: “Não acho que precisaria de ajuda de um técnico para usar o sistema.”

O sistema foi feito para ser de simples utilização, onde o usuário possa desfrutar de forma mais rápida e trivial possível. Para tanto, foi questionado aos participantes se eles conseguiriam utilizar o sistema sem a ajuda de um técnico. Nesse quesito obteve-se 78% de notas entre 5 e 4, 11% de notas 3, 4% de notas 2 e 7% pessoas discordavam completamente. Isso mostra que existe uma necessidade de melhorar o sistema, deixando-o ainda mais intuitivo, para que o número de pessoas que experimentarem essa dificuldade diminua cada vez mais.

A Figura 4.16 mostra as porcentagens sobre a afirmação 6.

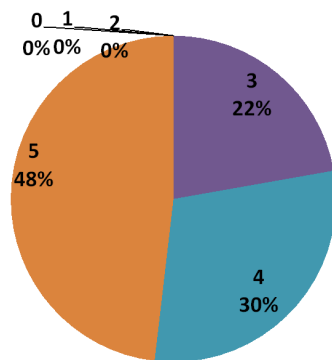


Figura 4.16: “Imagino que a maioria das pessoas aprenderia a utilizar o sistema rapidamente.”

Na mesma linha de pensamento, a afirmação 6 questiona a velocidade de aprendizado para a utilização do sistema. Nesse quesito temos, novamente, 78% de notas 5 e 4, porém os outros 22% limitaram-se a nota 3. Indício de que, por mais que o usuário sentisse que precisava de ajuda para a utilização do sistema, uma rápida explicação foi suficiente para ajudá-lo.

A afirmação acerca do interesse mercadológico (afirmação 7) tem suas porcentagens indicadas na Figura 4.17.

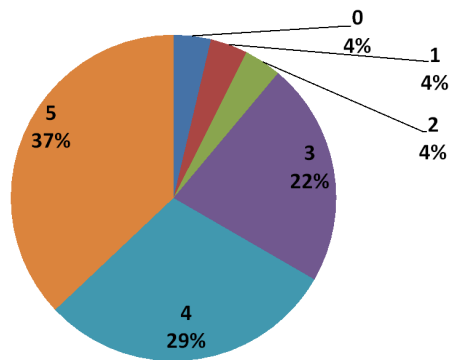


Figura 4.17: “Compraria um jogo que apresentasse essa funcionalidade.”

Para verificar a potencialidade mercadológica do trabalho realizado, os participantes deveriam avaliar se comprariam ou não um jogo que apresentasse a funcionalidade. 66% dos mesmos deram nota 5 e 4 para a questão 7, que apresentava este questionamento. 22% avaliaram com nota 3, e as notas 0, 1 e 2 foram dadas por 4% dos participantes cada. Este resultado indica que há uma necessidade de um estudo para verificar como aumentar o interesse pelo sistema, em termos mercadológicos.

A 8ª afirmação vem a questionar os participantes se dentro da situação ao qual foi proposto o sistema houve vantagens ao que já é utilizado no mercado. Seu gráfico de porcentagens é mostrado na Figura 4.18.

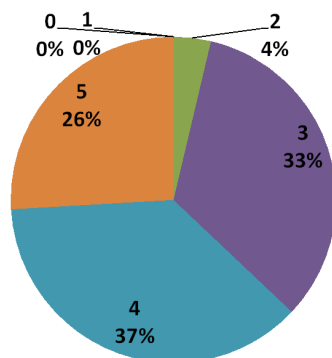


Figura 4.18: “Este sistema apresenta vantagens em relação ao modelo convencional de jogos de luta.”

Como pode-se observar: 63% dos participantes deram notas 5 e 4, 33% avaliou como 3 e 4% marcou 2. Inere-se uma avaliação positiva dentro do escopo do estudo e uma percepção de evolução tecnológica por parte dos participantes.

As porcentagens da 9ª questão seguem na Figura 4.19.

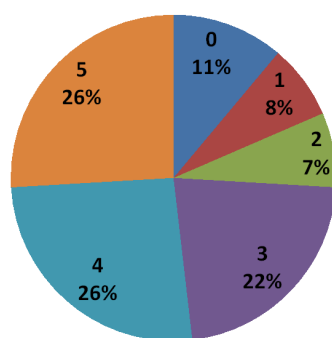


Figura 4.19: “Sou capaz de imaginar outras formas de utilizar o sistema de integração.”

Um sistema que não pudesse ser utilizado em outros projetos se tornaria logo obsoleto. Assim o participante foi convidado a imaginar outras formas de utilizar a interação *Kinect* e celular. Pretende-se garantir que não é apenas um jogo ou um tipo de jogo que pode se utilizar do sistema, tornando o mesmo uma alternativa real de implementação na área de jogos e uma solução inovadora para problemas de outras áreas. Obteve-se 26% de notas 5, 26% de notas 4, 22% de notas 3, 26% de notas entre 0, 1 e 2. Resultado que não indicou nenhuma tendência forte, talvez seja necessário focar em um grupo de estudo mais específico.

Por fim, a Figura 4.20 traz as porcentagens das notas da afirmação 10.

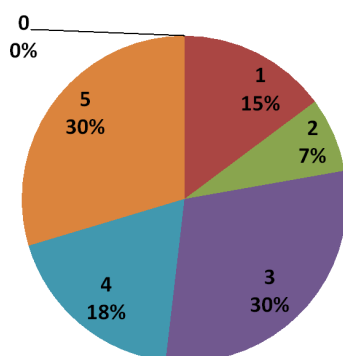


Figura 4.20: “Se tivesse de comprar um aparelho específico para a simulação dos movimentos da espada, seria um incômodo.”

A última afirmação avalia se a opção de utilizar o celular ao invés de outros aparelhos específicos é algo interessante ao público comprador. As notas 5 e 4 ficaram com 48% das avaliações, 30% dos participantes marcaram a nota 3 enquanto 22% marcaram 2 ou 1. Os resultados também se mostraram inconclusivos. Possivelmente comprar um novo dispositivo não seja um grande incômodo para 55% dos participantes da pesquisa.

Após a aquisição dos dados, foi de grande interesse a avaliação dos comentários. Seguem alguns exemplos:

“Poderia ter mais opções de movimentos”

“Dar aos lutadores a possibilidade de agachar ou até, se possível, usar o escudo como forma de atingir ao adversário”

“Jogo interessante, mas é necessário melhorar o tempo de resposta”

“Incluir instruções, melhorar o sincronismo para permitir que os movimentos sejam mais rápidos”

Os comentários deram indicações de qual deve ser o foco em trabalhos futuros: procurar aumentar o número de movimentos permitidos dentro do jogo e diminuir o atraso no tempo de resposta da espada.

Observados os dados da pesquisa, tem-se uma conclusão positiva do sistema implementado. No caso das afirmações 1, 5 e 6 que falam acerca de aspectos de familiarização do sistema, temos todas com mais de 70% de notas entre 4 e 5, indicando uma concordância dentre os participantes com a frase. É assim passível de conclusão que, dentro de uma amostra escolhida aleatoriamente, o sistema não teria rejeição por fatores de dificuldade, tornando a experiência de jogo mais real, agradável e natural.

As afirmações 3, 7 e 8 procuraram avaliar a qualidade da experiência dos participantes ao utilizar o sistema, observando o quanto a pessoa sentia prazer ao utilizar o jogo dotado do sistema, a intenção de compra e se via algum tipo de vantagem. Nessas afirmações a porcentagem de notas entre 4 e 5 foram respectivamente 93%, 66% e 53%, todas acima de 50%. Percebe-se que, pelo menos na amostra que a pesquisa utilizou, o jogo parece divertir o jogador e cria nele o interesse de compra. Já a visão de vantagens sobre os jogos atuais é inconclusiva.

Um sistema que possa ser utilizado em mais possibilidades do que foi proposto inicialmente, possivelmente terá maior penetração de mercado e maior utilização entre os usuários. Assim, as perguntas 4 e 9 avaliam se a utilização do *smartphone* com o *Kinect* é algo que não traz nenhum tipo de recusa do usuário final, se ele trata como uma possibilidade viável e se o mesmo é capaz de imaginar outras maneiras de utilização do conjunto. Temos sim uma tendência em concordância com a primeira, porém uma segunda informação inconclusiva.

Foi necessária uma avaliação do uso do celular como alternativa. Se seu tempo de resposta não fosse aceitável, haveria um desagrado do público e, portanto, não seria uma opção. Entretanto, houve 63% de notas favoráveis, porém em sua grande maioria na nota 4 e grande parte dos comentários escritos foram relacionados à qualidade da sincronização. Assim, pode-se dizer que o sincronismo está adequado ao jogo, porém não completamente satisfatório.

Por fim tentamos avaliar se a opção de não precisar adquirir uma nova peça de hardware seria encarada pelos participantes como uma vantagem. Os resultados mostraram

que não houve uma concordância dos participantes, tendo apenas 48% de notas entre 4 e 5. Assim, possivelmente o ato de comprar algo para a função de espada não seja incômodo para a maioria dos participantes da pesquisa.

Capítulo 5

Conclusão

Este trabalho apresentou a proposta e desenvolvimento de um sistema de integração entre dispositivos *smartphone* e o *Kinect* como uma nova interface de controle jogador/jogo. Foram utilizados a plataforma de desenvolvimento de jogos *Unity3D* e os *plugins GyroDroid* e *ZigFu* afim de fazer a integração dos sensores dos celulares e do *Kinect* com a plataforma *Unity3D*. Um protótipo de jogo foi implementado para avaliação da solução, onde dois jogadores podem encarnar dois gladiadores romanos dotados de espada e escudo e simularem lutas entre si. Uma pesquisa de opinião foi realizada com 27 participantes, afim de validar a solução proposta.

Os movimentos do corpo e do pulso direito foram rastreados com precisão. O jogo implementado alcançou as nossas expectativas e a pesquisa indica uma concordância por parte dos participantes no que diz respeito à validade do sistema.

A maior dificuldade percebida se deu na implementação do sincronismo da espada com o celular, porém, mesmo esta, parece não apresentar impedimento aos nossos participantes dentro do jogo proposto. Uma possível solução para tanto seria realmente integrar o *Wii Remote Plus* ao *Kinect*, porém o gasto se tornaria alto e a facilidade de inserção do sistema no mercado diminuiria dado que os fabricantes são concorrentes no mercado de jogos. Com o nosso sistema, qualquer *smartphone* dotado de giroscópio e acelerômetro é capaz de agir como rastreador de movimento da mão em conjunto com um *Kinect*.

Um possível trabalho futuro seria a integração do *Kinect 2.0* da *Microsoft*, que possui um desempenho melhor, de acordo com o fabricante, ou mesmo a utilização de uma *SmartTV* para tanto, tentando reduzir o número de aparelhos necessários para utilizar o sistema. Para a movimentação da mão, seria interessante implementar um *plugin* próprio de captura dos sensores do celular, de modo a procurar uma sensibilidade e precisão maiores, além de permitir o uso de dispositivos com outros sistemas operacionais. E por fim implementar um sistema de reconhecimento de voz, para tornar a imersão ainda maior dentro dos jogos que se utilizem da solução proposta.

Referências

- [1] Inertial Aerosystems. Hmc1512 two axis sensor. http://www.inertial.co.uk/catalog_details.asp?ProdID=396. vi, 18
- [2] Bruno Andrade. Hardware. <http://brunoaa.wordpress.com/2011/02/10/hardware/>, Fev 2011. vi, 19
- [3] Matej Andrejašic. Mems accelerometers. In *University of Ljubljana. Faculty for mathematics and physics, Department of physics, Seminar*, 2008. 19
- [4] Diego Bichara and Alexandre Henrique Baião. Demonstração de tg k++. <https://www.youtube.com/watch?v=fhcP6JjXrrE>. 28
- [5] Emily Brown and Paul Cairns. A grounded investigation of game immersion. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 1297–1300. ACM, 2004. 1
- [6] K. Brown. *Reflections on Relativity*. Lulu Enterprises Incorporated, 2011. 15
- [7] Frank Caron. Of gyroscopes and gaming: the tech behind the wii motionplus. <http://arstechnica.com/gaming/2008/08/wii-motion-sensor/>, Aug 2008. 15
- [8] Jason Chen. Microsoft xbox 360 kinect launches november 4. <http://gizmodo.com/5563148/microsoft-xbox-360-kinect-launches-november-4>. 14
- [9] Anthony Chu. Choosing the right type of accelerometers. vi, 17, 18
- [10] Fullbright Company. About gone home. <http://fullbright.company/gonehome/>. 20
- [11] Fernando de Castro Junqueira and Ettore Apolônio de Barros. Development of a dynamically tuned gyroscope–dtg. *ABCM Symposium Series in Mechatronics-Vol*, pages 470–478, 2004. 15
- [12] Michelle Ferreira. 63% dos brasileiros usam o smartphone em menos de 1 minuto após acordarem. <http://epocanegocios.globo.com/Inspiracao/Vida/noticia/2013/12/63-dos-brasileiros-usam-o-smartphone-em-menos-de-1-minuto-apos-acordarem.html>, Dez 2013. vi, 3, 4

- [13] 3D Focus. The world's first gyro-based finger tracking glove (video). <http://www.3dfocus.co.uk/3d-news-2/the-worlds-first-gyro-based-finger-tracking-glove-video/13567>, Jul 2013. vi, 5
- [14] Muse Games. Guns of icarus online. <http://musegames.com/games/guns-of-icarus-online/>, 2013. 20
- [15] V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun. Real time motion capture using a single time-of-flight camera. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 755–762, June 2010. vi, 10
- [16] Varun Ganapathi, Christian Plagemann, Daphne Koller, and Sebastian Thrun. Real-time human pose tracking from range data. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI, ECCV'12*, pages 738–751, Berlin, Heidelberg, 2012. Springer-Verlag. vi, 10, 11
- [17] Roberta Gouveia. O que é um acelerômetro capacitivo. <http://www.mecanicaindustrial.com.br/conteudo/136-o-que-e-um-acelerometro-capacitivo>, Fev 2012. vi, 17
- [18] Richard Isely III. Microsoft kinect development. University of Wisconsin Platteville. 14
- [19] Texas Instruments. Accelerometers and how they work. 2005. 16
- [20] iWaggle3D. iwatch | ninja gaiden 3 playstation move analysis. https://www.youtube.com/watch?v=7VU0g_NsYoA, Mar 2012. vi, 14
- [21] M. Joselli, J.R. da Silva, M. Zamith, E. Clua, M. Pelegriño, E. Mendonca, and E. Soluri. An architecture for game interaction using mobile. In *Games Innovation Conference (IGIC), 2012 IEEE International*, pages 1–5, Sept 2012. vi, 12
- [22] Byoung-Doo Kang, Kye-Hoon Jeon, Dongwuk Kyoung, Seung-Hun Kim, and Jung-Hoon Hwang. Multiple human body tracking based on normalized cross-correlation of average histogram using the fusion of color and thermal image sensor. In *RO-MAN, 2012 IEEE*, pages 398–403. IEEE, 2012. vi, 8, 9
- [23] M. Katzourin, D. Ignatoff, L. Quirk, J.J. LaViola, and O.C. Jenkins. Swordplay: Innovating game development through vr. *Computer Graphics and Applications, IEEE*, 26(6):15–19, Nov 2006. vi, 13
- [24] Pop Chart Lab. The evolution of video game controllers. <http://popchartlab.com/products/the-evolution-of-video-game-controllers>. vi, 2
- [25] Yun-Seok Lee, Dae-Keun Yoon, Jong-Tae Seo, and Byung-Ju Yi. Automatic tracking of human-body using a 3 dof robotic lamp. In *Mechatronics and Automation (ICMA), 2011 International Conference on*, pages 470–476. IEEE, 2011. vi, 9

- [26] Jan-Cheng Lin and Cheng-Ming Huang. 3d hand posture tracking with depth gradient estimation on a rgb-d camera. In *Consumer Electronics (ISCE), 2013 IEEE 17th International Symposium on*, pages 109–110. IEEE, 2013. vi, 7
- [27] Dimension Engineering LLC. A beginner’s guide to accelerometers. <http://www.dimensionengineering.com/info/accelerometers>. 16
- [28] S.M. Malfatti, F. Ferreira dos Santos, and S. Rodrigues dos Santos. Using mobile phones to control desktop multiplayer games. In *Games and Digital Entertainment (SBGAMES), 2010 Brazilian Symposium on*, pages 230–238, Nov 2010. vi, 11, 12
- [29] MEGGIT. Model 2262a. Mai 2011. vi, 17
- [30] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BMVC*, pages 1–11, 2011. vi, 6
- [31] Iasonas Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2088–2095. IEEE, 2011. vi, 7
- [32] prefrontal cortex. Gyrodroid. <https://www.assetstore.unity3d.com/en/#!/content/1701>, Jul 2014. 20
- [33] quangng. Hall effect sensing principle. <https://wiki.metropolia.fi/display/sensor/Hall+effect+sensing+principle>, Mar 2014. vi, 18
- [34] Ganesh Rakate. Human body tracking system based on dwt and mean-shift algorithm on arm-linux platform. In *Information and Communication Technologies (WICT), 2012 World Congress on*, pages 715–720. IEEE, 2012. vi, 8
- [35] Amalia Eka Rakhmania. Accelerometers and how they work. <http://ichbinnia.wordpress.com/2012/12/31/acceleration-sensor/>, December 2012. vi, 18
- [36] Javier Romero, Hedvig Kjellström, Carl Henrik Ek, and Danica Kragic. Non-parametric hand pose estimation with object context. *Image Vision Comput.*, 31(8):555–564, August 2013. vi, 6
- [37] Sensors and Actuators. How accelerometer works. <http://sensors-actuators-info.blogspot.com.br/2012/02/accelerometer.html>, Fev 2012. 17
- [38] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '11*, pages 1297–1304, Washington, DC, USA, 2011. IEEE Computer Society. 14
- [39] Behold Studios. Knights of pen and paper. <http://beholdstudios.com.br/knights-of-pen-paper/>, Oct 2012. 20

- [40] V. Tam and Ling-Shan Li. Integrating the kinect camera, gesture recognition and mobile devices for interactive discussion. In *Teaching, Assessment and Learning for Engineering (TALE), 2012 IEEE International Conference on*, pages H4C–11–H4C–13, Aug 2012. vi, 12, 13
- [41] Cheng Tang, Yongsheng Ou, Guolai Jiang, Qunqun Xie, and Yangsheng Xu. Hand tracking and pose recognition via depth and color information. In *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*, pages 1104–1109. IEEE, 2012. vi, 7, 8
- [42] Solid State Technology. Introduction to mems gyroscopes. <http://electroiq.com/blog/2010/11/introduction-to-mems-gyroscopes/>, Nov 2010. vi, 15, 16
- [43] Robert Wang, Sylvain Paris, and Jovan Popović. 6d hands: Markerless hand-tracking for computer aided design. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 549–558, New York, NY, USA, 2011. ACM. vi, 5, 6
- [44] Zigfu. Zdk for unity3d. <http://zigfu.com/en/zdk/unity3d/1>. 20