

UNIVERSIDADE ESTADUAL PAULISTA

“Júlio de Mesquita Filho”

Programa de Pós-Graduação em Ciência da Computação

Isabele Andreoli Agostinho

Combinação de Dispositivos de Baixo Custo para
Rastreamento de Gestos

UNESP

2014

Isabele Andreoli Agostinho

Combinação de Dispositivos de Baixo Custo para
Rastreamento de Gestos

Orientador: Prof. Adj. José Remo Ferreira Brega

Dissertação de Mestrado elaborada junto ao Programa de Pós-Graduação em Ciência da Computação – Área de concentração em Computação Aplicada, como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação.

Data de apresentação: 18 de fevereiro de 2014.

Resultado: Aprovada

Banca examinadora

José Remo Ferreira Brega, Prof. Adj.
Universidade Estadual Paulista “Júlio de Mesquita Filho”

Ildeberto Aparecido Rodello, Prof. Dr.
Universidade de São Paulo

Aparecido Nilceu Marana, Prof. Adj.
Universidade Estadual Paulista “Júlio de Mesquita Filho”

UNESP

2014

Agradecimento

Agradeço a Deus, por Seu amor, inspiração e ter me conduzido até aqui.

Ao Professor Doutor José Remo Ferreira Brega, pela oportunidade de desenvolver esta pesquisa, por ter acreditado em mim, pelos ensinamentos, orientações, paciência e otimismo.

A minha mãe, por todo encorajamento, compreensão e apoio incondicional.

Aos colegas do Laboratório de Sistemas de Tempo Real, Mario e Diego, pelos auxílios e dicas valiosas.

Aos colegas de trabalho, por todo incentivo e apoio recebido.

“Consagre ao Senhor tudo o que você faz, e os seus planos serão bem-sucedidos.”

Provérbios 16:3

Sumário

Lista de Ilustrações.....	vi
Lista de Tabelas.....	ix
Lista de Abreviaturas	x
Lista de Códigos.....	xi
Resumo.....	xii
Abstract	xiii
1 Introdução	1
1.1 Justificativa	2
1.2 Objetivos	3
1.3 Estrutura da Dissertação.....	3
2 Rastreamento de movimentos	5
2.1 Definições	5
2.2 Gestos	6
2.3 Dispositivos de rastreamento	7
2.4 Aplicações em rastreamento.....	12
2.5 Tecnologias de rastreamento dos dispositivos utilizados no sistema	13
2.5.1 Tecnologia ótica	13
2.5.1.1 Elementos Básicos.....	14
2.5.1.2. Infravermelho	15
2.5.2 Tecnologia inercial.....	17
2.5.3 Tecnologias usadas em luvas de dados	18
2.6 Considerações em relação ao rastreamento.....	20
2.6.1 Calibração.....	20
2.6.2 Previsão	21
2.6.3 Parâmetros de desempenho	22
2.6.4 Rastreamento com visão computacional	23
2.6.5 Rastreamento com sensores inerciais e luvas de dados	24
2.7 Combinação de tecnologias.....	25
2.8 Considerações finais.....	26
3 Dispositivos Comerciais de Rastreamento	28

3.1 Kinect.....	28
3.1.1 Hardware.....	29
3.1.2 Kinect SDK.....	30
3.2 Wii Remote.....	31
3.2.1 Hardware.....	32
3.2.2 Wii Remote SDK.....	32
3.3 Luva.....	33
3.3.1 Hardware.....	34
3.3.2 5DT Data Glove Ultra SDK.....	35
3.4 Considerações finais.....	37
4 Sistema proposto e desenvolvido.....	39
4.1 Módulos.....	42
4.1.1. 5DT Data Glove Ultra.....	42
4.1.2 Wii Remote.....	47
4.1.3 Kinect.....	51
4.1.4 Protótipo de Humano Virtual.....	54
4.1.5 Módulo Principal.....	57
4.2 Sincronização.....	58
5 Resultados.....	64
5.1 Resultados da luva.....	64
5.2 Resultados do Wii Remote.....	65
5.3 Resultados do Kinect.....	67
5.4 Resultados da combinação.....	69
6 Conclusão.....	76
Referências.....	79
ANEXO A – Tecnologias e pesquisas com GCUI.....	86

Lista de Ilustrações

Figura 2.1 – Dispositivo para rastreamento magnético	09
Figura 2.2 – Dispositivo mecânico de rastreamento	09
Figura 2.3 – Dispositivos com marcadores para reconhecimento ótico	10
Figura 2.4 – Luva com sensor para rastreamento com ultra-sons	10
Figura 2.5 – Exemplo de uso de acelerômetros para rastreamento	11
Figura 2.6 - Os seis graus de liberdade possíveis em sistemas de rastreamento e sua representação para movimentação da cabeça	11
Figura 2.7 – Espectro eletromagnético	14
Figura 2.8 – Dispositivo de rastreamento inercial comercializado pela Xsens, composto por giroscópio, acelerômetro e magnetômetro	17
Figura 2.9 – Luva Haptic Telexistence	18
Figura 2.10 – Luva com marcadores óticos reflexivos	18
Figura 2.11 – Elementos de uma luva com fibra ótica	19
Figura 2.12 – Luva da 5DT com sensores de fibra ótica	19
Figura 2.13 – Esquema de uma fibra ótica	20
Figura 2.14 – Sistema de rastreamento usando sensores de ultra-sons (Hexamite) e inerciais (Xsens)	25
Figura 3.1 – Sensores óticos do Kinect.....	30
Figura 3.2 - Estrutura do OpenNI.....	31
Figura 3.3 – Sistema de coordenada do Wii Remote.....	32
Figura 3.4 – Esquema da 5DT Data Glove Ultra.....	34
Figura 3.5 – Representação dos sinais capturados pela 5DT Data Glove Ultra.....	35

Figura 3.6 – Ilustração dos gestos.....	36
Figura 4.1 – Exemplos de erro no rastreamento feito pelo Kinect devido a oclusões.....	39
Figura 4.2 – Proposta de combinação para rastreamento de gestos.....	40
Figura 4.3 - Diagrama de módulos do Sistema.....	41
Figura 4.4 - Organização dos arquivos de acesso às luvas.....	43
Figura 4.5 - Esquema de funcionamento do módulo da 5DT Data Glove.....	46
Figura 4.6 - Interface gráfica para controle das luvas.....	47
Figura 4.7 - Posicionamento do Wii Remote.....	48
Figura 4.8 - Diagrama de atividades do módulo de rastreamento pelo Wii Remote.....	49
Figura 4.9 - Interface gráfica para interação com os controles Wii Remote.....	50
Figura 4.10 - Diagrama do módulo de rastreamento pelo Kinect.....	53
Figura 4.11 - Interface do módulo do Kinect.....	54
Figura 4.12 - Grafo de cena do ambiente.....	56
Figura 4.13 - Grafo de cena do avatar.....	57
Figura 4.14 – Interface gráfica do módulo principal do sistema.....	58
Figura 4.15 – Diagrama de atividades do sistema.....	59
Figura 4.16 – Esquema de sincronização dos dados do sistema.....	62
Figura 5.1 – Teste de movimentação do Avatar com a luva.....	65
Figura 5.2 – Teste de rotação dos antebraços.....	66
Figura 5.3 – Primeiro teste do Kinect.....	67
Figura 5.4 – Segundo teste do Kinect.....	68
Figura 5.5 – Teste da combinação dos dispositivos dedilhando a mão direita.....	70

Figura 5.6 – Teste da combinação dos dispositivos com movimentos circulares das mãos em frente ao corpo, com as palmas voltadas para baixo.....	71
Figura 5.7 – Teste da combinação dos dispositivos fechando a mão junto ao corpo.....	72
Figura 5.8 – Teste da combinação dos dispositivos movendo as mãos da esquerda para a direita com as palmas para cima.....	73
Figura 5.9 – Teste da combinação dos dispositivos para as poses estáticas indicando a letra “L” e “T” em LIBRAS, polegar estendido nas duas mãos, e apenas na mão esquerda, voltado para baixo e com o braço esticado.....	74

Lista de Tabelas

Tabela 2.1 - Vantagens e desvantagens das tecnologias de rastreamento ótico, inercial e por luva de dados.....	27
Tabela 3.1 – Índices dos sensores para a 5DT Data Glove Ultra.....	34
Tabela 3.2 – Esquema de definição de gesto implementado pelo SDK da 5DT Data Glove Ultra.....	36
Tabela 3.3 – Comparativo entre os dispositivos.....	38
Tabela A.1 - Projetos comerciais com tecnologia voltada à GCUI.....	86
Tabela A.2 - Pesquisas envolvendo tecnologia GCUI.....	86

Lista de Abreviaturas

3D	Três Dimensões
API	Application Programming Interface
AV	Ambiente Virtual
CCD	Charge-Coupled Devices
DOF	Degrees of Freedom
DSC	Digital Static Camera
FPS	Frame Per Second
GCUI	Gesture Controlled User Interface
HMM	Hidden Markov Models
HV	Humano Virtual
IHC	Interação Humano-Computador
IR	Infrared
JNI	Java Native Interface
LED	Light Emitting Diodes
LGPL	GNU Lesser General Public License
LIBRAS	Língua Brasileira de Sinais
OPENNI	Open Natural Interface
RGB	Red Green Blue
RV	Realidade Virtual
SDK	Software Development Kit
USB	Universal Serial Bus

Lista de Códigos

Código 4.1 – Método para conectar luvas presente no sistema.....	44
Código 4.2 – Carregamento da biblioteca da luva e assinatura do método de conexão das luvas.....	44
Código 4.3 – Chamada de método pelo JNI em JDataGlove5DT.cpp.....	44
Código 4.4 – Método de conexão das luvas em DataGlove5DT.cpp.....	45
Código 4.5 – Método para obter os ângulos das juntas a partir da matriz de rotação.....	52
Código 4.6 – Método para converter valores do Wii Remote do antebraço esquerdo para o sistema de eixos do Avatar impondo limite de movimentos.....	60
Código 4.7 – Obtenção, tratamento e envio de dados por KinectUserTraker.....	62
Código 4.8 – Definição do movimento a ser reproduzido no Avatar.....	63
Código 4.9 – Métodos de TrackerInterfaceAvatar usados para a pré-renderização do HV.....	63

Resumo

Algumas pesquisas mostram que a combinação de mais de uma tecnologia de sensor pode melhorar o rastreamento de movimentos, tornando-o mais preciso ou permitindo a implementação de aplicações que usam movimentos complexos, como nas línguas de sinais por exemplo. A combinação de dispositivos de rastreamento de movimentos vendidos comercialmente permite desenvolver sistemas de baixo custo e de fácil utilização. O Kinect, o Wii Remote e a 5DT Data Glove Ultra são dispositivos que usam tecnologias que fornecem informações complementares de rastreamento de braços e mãos, são fáceis de usar, têm baixo custo e possuem bibliotecas de desenvolvimento gratuitas, entre outras vantagens. Para avaliar a combinação desses dispositivos para rastreamento de gestos, foi desenvolvido um sistema de rastreamento que contém dois módulos principais, um de tratamento dos dispositivos, com inicialização e junção dos movimentos, e outro com a visualização da movimentação do Humano Virtual para o rastreamento feito. Este sistema utiliza a luva para a captura da configuração das mãos, o Wii Remote para fornecer a rotação dos antebraços e o Kinect para o rastreamento dos braços e da inclinação dos antebraços. Foram executados testes para vários movimentos, e os resultados obtidos relativos a cada dispositivo foram tratados e o rastreamento reproduzido em tempo real no Humano Virtual com sucesso.

Palavras-chave: dispositivos de rastreamento, rastreamento de gestos, combinação de dispositivos.

Abstract

Some researches show that combination of more than one sensor technology can improve tracking, making it more precise or making possible the development of systems that use complex movements, such as in sign languages. The combination of commercial tracking devices allows the development of low cost and easy to use systems. The Kinect, the Wii Remote and the 5DT Data Glove Ultra are devices that use technologies that give complementary information of arms and hands tracking, are easy to manipulate, have low cost and free development tools, among other advantages. To evaluate the combination of these devices for human communication gesture tracking, a system was developed having two main modules, one for device processing with initialization and movements union, and other that provides the visualization of Virtual Human movements of executed tracking. This system use the glove to provide hands configuration capture, the Wii Remote to give forearms rotation and a Kinect to track arms and forearms pitch. Tests were done for different movements, and the results of each devices data were processed and tracked movement was displayed by the Virtual Human in real time successfully.

Key words: tracking devices, gesture tracking, sign language, devices combining.

1 Introdução

Os diferentes meios de entrada de dados em sistemas de Realidade Virtual (RV) permitem o desenvolvimento de aplicações computacionais antes impossíveis, como a geração de gestos por meio do rastreamento de movimentos. Porém, conforme levantamento bibliográfico realizado, as pesquisas na área ainda buscam resolver problemas como falta de precisão, complexidade de uso e/ou alto custo dos sistemas existentes, dificultando o amplo uso de aplicações que utilizam o rastreamento de gestos como meio de entrada.

Essas pesquisas apresentam diferentes dispositivos e tecnologias desenvolvidas visando a solução desses problemas. Como consequência, novos meios de interação com sistemas computacionais têm sido criados.

A Interação Humano-Computador (IHC) realizada em sistemas de RV é baseada no uso de dispositivos conhecidos como não convencionais. Diferentes tipos de dispositivos foram agregados aos sistemas de RV ao longo dos anos, tais como rastreadores magnéticos e luvas de dados (BURDEA; COIFFET, 2003). Os dispositivos não convencionais empregam tecnologias variadas para seu funcionamento e a evolução delas permitiu a popularização e ampliação de seu uso em sistemas diversos, nem sempre com a finalidade para a qual foram originalmente propostos.

Cada tecnologia possui suas particularidades e apresenta melhor desempenho em situações específicas. Várias pesquisas já foram feitas (BHUIYAN e PICKING, 2009; ZHOU e HU, 2004), e continuam sendo, no sentido de determinar quais tecnologias se ajustam melhor aos requisitos de diferentes aplicações. O rastreamento de movimentos é um tipo de aplicação que tem sido feita com diversos tipos de sensores diferentes. Cada um apresenta vantagens e desvantagens de acordo com o objetivo do rastreamento (BHUIYAN; PICKING, 2009).

A combinação de mais de um dispositivo, com tecnologias diferentes, foi utilizada em alguns trabalhos visando o aprimoramento do rastreamento. Essa combinação tem tornado possível rastreamentos complexos e o aumento de precisão das aplicações.

Entretanto, os principais dispositivos utilizados nesses trabalhos são específicos para a aplicação dos trabalhos analisados. São configurados de maneira a se tornarem

dispositivos especializados, para serem usados por pessoas com conhecimento no sistema fruto daquela pesquisa. Existem vários dispositivos comerciais para uso geral, mas costumam ser caros e difíceis de serem adquiridos.

Dentre as opções disponíveis atualmente no mercado, três dispositivos se destacaram por preencherem os requisitos de facilidade de uso e baixo custo: o Kinect (KINECT, 2011), o Wii Remote (NINTENDO, 2011) e a luva 5DT Data Glove Ultra (FIFTH DIMENSION TECHNOLOGY – 5DT, 2011).

1.1 Justificativa

O reconhecimento de gestos complexos, que exigem o rastreamento dos braços, antebraços, posição das mãos e flexão dos dedos, como os utilizados para comunicação humana por língua de sinais, por exemplo, pode ser utilizado em diversos tipos de aplicação, como ferramentas inclusivas para deficientes auditivos, treinamento de atividades de risco, manipulação de objetos delicados, automatização da geração de gestos em jogos, entre outros. Mas este tipo de rastreamento ainda pode ser aperfeiçoado para se tornar mais preciso e acessível, tanto financeiramente quanto para usuários leigos nas tecnologias envolvidas.

No levantamento bibliográfico realizado, percebeu-se uma escassez de trabalhos voltados ao rastreamento dos movimentos dos braços, mãos e dedos, usando mais de um tipo de sensor.

Existem sistemas de rastreamento que utilizam uma só tecnologia, como sistemas de visão computacional com duas ou mais câmeras, capazes de minimizar os problemas de oclusão e captar os movimentos das mãos (GHIROTTI; MORIMOTO, 2010), ou sistemas que utilizam apenas dispositivos inerciais e óticos para rastreamento do corpo inteiro (NGUYEN et al. , 2011), assim como sistemas que usam apenas dispositivos inerciais (DONG; WU; CHENG, 2007).

Estas soluções, entretanto, exigem que os sensores sejam colocados em locais específicos, com distâncias e ângulos entre si predeterminadas de acordo com o *software* desenvolvido. Permitem também o uso de *hardwares* diversos de sensores com o mesmo fim, não padronizando a configuração do sistema.

Nenhum dos sistemas encontrados visava o rastreamento de baixo custo, de fácil utilização e com precisão suficiente para rastrear gestos complexos como os de línguas de sinais. Em vários casos, o uso do sistema era restrito a ambientes controlados, como laboratórios e estúdios, visando minimizar a ocorrência de interferências que prejudicassem o rastreamento.

A combinação dos equipamentos escolhidos para este trabalho exige apenas que o Kinect seja colocado a uma distância dentro de sua faixa de alcance.

O uso de outros dispositivos pode se tornar menos intuitivo com a aplicação de equipamentos específicos que não são de conhecimento popular como o Wii Remote e o Kinect. Outras desvantagens são a exigência de *hardwares* adicionais, a eliminação do aproveitamento dos softwares já existentes para os dispositivos e o custo. Enquanto o custo total aproximado dos equipamentos do sistema proposto é de cinco mil reais, outros equipamentos disponíveis no mercado, como o MotionStar e o Fastrak, podem custar dezenas de milhares de dólares (ASCENSION, 2013; POLHEMUS, 2013).

1.2 Objetivos

O presente trabalho tem como objetivos apresentar e avaliar a viabilidade da combinação do Kinect, Wii Remote e a 5DT Data Glove Ultra para a captura de movimentos gestuais complexos. Essa combinação é proposta visando obter um rastreamento mais preciso, pela complementação das informações de movimento capturadas, com uma ferramenta que prioriza a facilidade de uso e seja de baixo custo, que tenha boa mobilidade e cujo uso não seja restrito a ambientes controlados, permitindo a visualização do movimento captado por meio de sua reprodução com o uso de um Humano Virtual (HV).

1.3 Estrutura da Dissertação

Esta dissertação está estruturada do seguinte modo. No Capítulo 2 discorre-se sobre alguns tópicos da IHC em sistemas de RV, dispositivos e aplicações em rastreamento,

apresentam-se alguns aspectos do funcionamento das tecnologias presentes no Kinect, no Wii Remote, e na 5DT Data Glove Ultra, fazem-se considerações sobre o rastreamento de movimentos usando as tecnologias dos dispositivos selecionados e apresentam-se trabalhos de pesquisa que exemplificam as aplicações de rastreamento, e o uso conjunto de mais de um tipo de dispositivo é discutido. No Capítulo 3 discorre-se sobre os dispositivos escolhidos para a implementação do sistema proposto, descrevendo-se o hardware de que são compostos e as bibliotecas de desenvolvimento. No Capítulo 4 detalha-se o desenvolvimento do trabalho. Nos Capítulos 5 e 6 apresentam-se os resultados e as conclusões, respectivamente.

2 Rastreamento de movimentos

A Realidade Virtual tem gerado técnicas e ferramentas que são utilizadas em diferentes áreas. Vários dispositivos de Interação Humano Computador que são conhecidos como sendo dispositivos de RV, estão sendo incorporados em sistemas de áreas diversas e dispositivos eletrônicos cujo uso tem se popularizado nos últimos anos.

Neste Capítulo será apresentada uma definição de RV e métodos de interface de RV. Após, aspectos de dispositivos de rastreamento para entrada de dados são discutidos. As tecnologias dos dispositivos usado neste trabalho são detalhadas. Em seguida, é apresentada uma definição de rastreamento no contexto da IHC, alguns aspectos de sistemas de rastreamento são apresentados, assim como os principais parâmetros de desempenho das tecnologias ótica, inercial e luva de dados. Exemplos de sistemas de rastreamento com cada uma dessas tecnologias são apresentados, de acordo com trabalhos presentes na literatura. Por fim, algumas combinações de tecnologias de rastreamento são apresentadas.

2.1 Definições

Não existe uma definição formal de Realidade Virtual mundialmente aceita. Diferentes autores usam diferentes definições (SHERMAN; CRAIG, 2003, p. 6). Em (TORI; KIRNER; SISCOUTO, 2006, p. 7) a RV é descrita como sendo “[...] uma interface avançada para aplicações computacionais, que permite ao usuário a movimentação (navegação) e interação em tempo real, em um ambiente tridimensional, podendo fazer uso de dispositivos multisensoriais, para atuação ou feedback”. Em (SHERMAN; CRAIG, 2003, p. 13), a RV é definida como “[...] um ambiente composto de simulações computacionais interativas que capta a posição e ações do participante e substitui ou acrescenta uma resposta para um ou mais sentidos, dando a sensação de estar mentalmente imerso ou presente dentro da simulação (um mundo virtual)” (tradução nossa).

Os sistemas de RV possuem métodos diferenciados de interface com o usuário. Os métodos convencionais de entrada e saída de dados, como o uso de teclado, mouse e monitores, também estão presentes em diversos sistemas de RV, mas são os dispositivos

diferenciados que potencializam a sensação de imersão, o envolvimento e a naturalidade na interação.

Por diferenciados pode-se entender que são dispositivos cujo uso com sistemas computacionais é mais recente que os métodos tradicionais, e que são voltados para uso em sistemas específicos com o objetivo de tornar a interação mais natural e próxima da realidade. Certas aplicações só são possíveis graças aos dispositivos e técnicas para entrada e saída de dados de RV. Podem ser citados como exemplos desses dispositivos diferenciados os sensores óticos, inerciais ou magnéticos que permitem o rastreamento de movimentos do usuário.

Sherman e Craig (2003, p. 88) definem rastreamento do corpo como sendo “(..)a habilidade do sistema de RV de perceber a posição e ações dos participantes” (tradução nossa). Os autores afirmam ainda que um sistema de rastreamento guarda informações sobre a movimentação do usuário ao longo do tempo, e que movimentos específicos podem ser chamados de gestos.

2.2 Gestos

Uma importante aplicação do rastreamento em sistemas computacionais é para o reconhecimento de gestos. Estes, além de serem ferramentas de interação com o computador, representando comandos de forma mais intuitiva, são comumente usados na interação entre as pessoas, como método de comunicação ou linguagem.

Segundo Sherman e Craig (2003), o gesto pode ser definido como um movimento específico do usuário que ocorre ao longo de um período de tempo. Quek (1995) criou uma classificação dos movimentos das mãos e braços separando-os em movimentos não intencionais e gestos, sendo que os gestos podem ser manipulativos ou comunicativos, e estes últimos são divididos em ações e símbolos.

O rastreamento e reconhecimento de gestos podem ser empregados para vários fins, como por exemplo, navegação e manipulação de ambientes virtuais, monitoramento do nível de atenção de motoristas, e reconhecimento de linguagem de sinais, entre outros. O reconhecimento automático de gestos requer a segmentação temporal do movimento, muitas

vezes utilizando como recurso a especificação de pontos de início e fim do gesto, como referência. Um mesmo gesto pode ter significados diferentes dependendo das ações efetuadas antes e depois dele, e da cultura onde o usuário está inserido. Geralmente, o significado do gesto depende da informação do espaço onde ele ocorre, da direção em que é feito, do sinal feito e da sua característica emocional, se houver (MITRA; ACHARYA, 2007). Esses autores fornecem de forma mais detalhada as técnicas mais comumente usadas para o rastreamento de mãos e braço, e face e cabeça.

Bhuiyan e Picking (2009) fornecem um histórico sobre a interação por meio de gestos. Os autores fornecem um comparativo dos projetos de pesquisa envolvendo Interface de usuário controlada por gestos (GCUI – *Gesture Controlled User Interface*) e dos projetos comerciais com tecnologia voltada para a GCUI, que podem ser visualizados nas tabelas do Anexo A.

Ghirotti e Morimoto (2010) fornecem outra definição e classificação de gestos e um exemplo de ferramenta que os emprega para IHC. Também definem que a estrutura de um GCUI pode ser dividida em segmentação, reconhecimento e interpretação.

Os gestos usados para interface possuem algumas diferenças com relação aos gestos feitos em linguagens de sinais: costumam ser específicos por aplicação (cada uma possui um conjunto próprio); esses conjuntos são relativamente pequenos em comparação com a quantidade de gestos de uma linguagem de sinais; e a complexidade do gesto é menor, envolvendo menos partes do corpo (SEGEN; KUMAR, 1998; YOON et al., 1999; GHIROTTI E MORIMOTO, 2010; DAVIS; SHAH, 1994).

2.3 Dispositivos de rastreamento

Os dispositivos tradicionais, tais como teclado e mouse, restringem os modos de interação com sistemas computacionais (PINHANEZ, 2006). Além disso, para se chegar o mais próximo possível da realidade, os sistemas de RV precisam fornecer métodos de IHC que sejam naturais ao ser humano. Esses métodos definem o grau de imersão do sistema de RV. Quanto maior a simulação de realidade, maior o grau de imersão.

Sherman e Craig (2003, p. 75 a 77) definem dois tipos de entrada em aplicações de RV: o monitoramento do usuário e o monitoramento do mundo com o qual o usuário interage. O primeiro tipo é o que torna possível ao sistema ser interativo, e requer um meio de rastrear alguma parte do corpo do usuário (entrada passiva), assim como as ações iniciadas pelo mesmo (entrada ativa). O segundo tipo fornece dados do mundo monitorado para a atualização do mundo virtual.

Pimentel e Teixeira (1995) classificam os dispositivos, de acordo com essa divisão, em dispositivos de trajetória (passivo) e de interação (ativo). A entrada passiva, foco de interesse deste trabalho, realizada através de técnicas de rastreamento de posições, faz uso de sensores de posição.

Os sensores de posição podem ser definidos como “[...] dispositivos que informam sua localização e/ou orientação ao computador. Geralmente, há uma peça fixa em uma posição previamente conhecida e uma ou mais unidades adicionais presas ao objeto sendo rastreado. [...] O sensor de posição é tido como o dispositivo de rastreamento mais importante de qualquer sistema de RV” (SHERMAN; CRAIG, 2003, p. 77, tradução nossa).

Vários autores definem diferentes tipos para tecnologias de rastreamento passivo (BURDEA; COIFFET, 2003; SHERMAN; CRAIG, 2003; TORI; KIRNER; SISCOUTO, 2006):

a) Magnética: fornece rastreamento preciso, mas sujeita a interferência de objetos metálicos no ambiente. Boa para ambiente controlado. Consiste em um emissor de campo magnético no ambiente, e receptores colocados no alvo a ser rastreado (Figura 2.1). Esses receptores medem o campo magnético captado, determinando a posição do alvo com relação ao emissor.



Figura 2.1 – Dispositivo para rastreamento magnético. Fonte: (ASCENSION TECHNOLOGY CORPORATION, 2013).

b) Mecânica: envolve a utilização de dispositivos mecânicos compostos por uma base onde o mecanismo é afixado, sendo a posição inicial do dispositivo em relação a ela previamente conhecida, a partir de onde é calculado o movimento efetuado no dispositivo pelo usuário. Geralmente são caras e de difícil mobilidade (Figura 2.2).



Figura 2.2 – Dispositivo mecânico de rastreamento. Fonte: (BLUTEAU et al., 2008).

c) Ótica: usados em sistemas de rastreamento chamados de sistema de visão computacional. Rastreia em duas ou três dimensões com uma ou mais câmeras. Este rastreamento está sujeito a oclusão e possui campo de atuação restrito, mas é barato, altamente portátil e não depende de equipamento especializado. Pode ser feito com o uso de marcadores na cena para facilitar a localização do objeto rastreado (Figura 2.3).



Figura 2.3 – Dispositivos com marcadores para rastreamento ótico. Fonte: (TEICHRIEB; FIGUEIREDO, 2010).

d) Ultra-sons: envolve três ou mais alto-falantes no ambiente e três microfones no objeto a ser rastreado. Está sujeito a oclusão e interferências de sons no ambiente. Tem baixo custo (Figura 2.4).



Figura 2.4 – Luva com sensor para rastreamento com ultra-sons. Fonte: (VRDEPOT, 2011).

e) Inercial: sensores eletromecânicos colocados no objeto cujo movimento se deseja rastrear, que medem aceleração e inclinação (Figura 2.5). Fornece medidas relativas, portanto passível de acumular erros no sistema e resultar em informações imprecisas. Isto pode ser reduzido com o uso de múltiplos sensores. Pode-se também fazer a recalibração manual periodicamente. Não possuem restrição de campo de atuação, nem precisam de unidades externas auxiliares, como no caso do rastreamento por ultra-sons ou eletromagnética. Os sensores mais comuns desse tipo são o acelerômetro e o giroscópio.

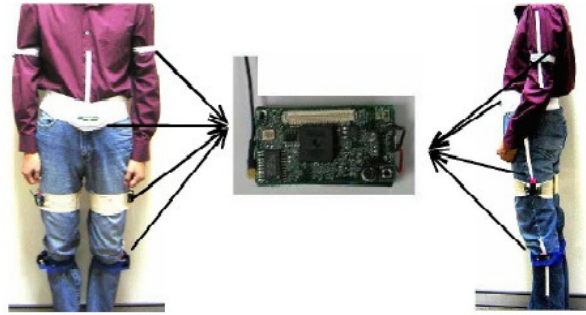


Figura 2.5 – Exemplo do uso de acelerômetros para rastreamento. Fonte: (DONG; WU; CHENG, 2007).

Cada classe de sensores possui suas vantagens e desvantagens, assim como diferentes graus de liberdade (DOF, do inglês, *Degrees of Freedom*), ou seja, cada sensor possui variada capacidade de rastreamento em diferentes direções, tendo como base os três eixos cartesianos (TORI; KIRNER; SISCOUTO, 2006). A Figura 2.6 ilustra este conceito.

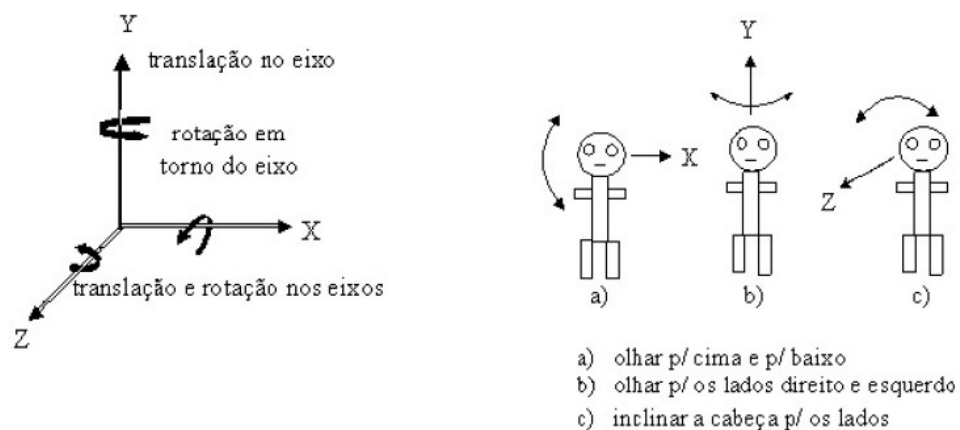


Figura 2.6 – Os seis graus de liberdade possíveis em sistemas de rastreamento e sua representação para movimentação da cabeça. Fonte: (TORI; KIRNER; SISCOUTO, 2006).

Existem três aspectos conflitantes em sistemas com rastreamento de posição: a precisão e velocidade do sensor utilizado, interferências no ambiente (metais, objetos opacos etc.), e a dificuldade de movimentação gerada pelo uso do dispositivo (peso, fios etc.). Não existem tecnologias de rastreamento capazes de satisfazer igualmente esses três aspectos (SHERMAN; CRAIG, 2003). Para escolher a tecnologia a ser usada quando do desenvolvimento de um sistema de RV, os requisitos referentes aos aspectos citados devem ser levados em consideração, de modo a priorizar os dispositivos que estejam melhores adaptados aos aspectos de maior relevância.

2.4 Aplicações em rastreamento

Diversas aplicações dependem do uso de dispositivos de RV para entrada de dados por meio de rastreamento do movimento do corpo humano. Aplicações que necessitem das informações sobre movimentação dos braços, antebraços e configuração das mãos, dependem do uso de tecnologias de entrada passiva. São exemplos dessas aplicações: o rastreamento de gestos de uma linguagem de sinais, de gestos para serem utilizados por personagens de um jogo de computador, movimentos usados em atividades de montagem, manutenção, treinamento ou manipulação de objetos, entre outras atividades.

Tais tipos de aplicação podem fazer uso de uma combinação de tecnologias para captar os movimentos, já que cada tipo de sensor possui pontos fortes e fracos, para satisfazer aos requisitos de rastreamento mínimo, de capturar a posição e orientação dos braços, antebraços, mãos e dedos.

Na linguagem de sinais, por exemplo, uma palavra é composta pela posição e orientação das mãos, dedos, gestos realizados por estes em conjunto com os braços, assim como o local do corpo ou do espaço onde este gesto é realizado (LIBRAS.ORG.BR, 2011). Um sistema de reconhecimento de linguagem de sinais possui, portanto, um espaço bem definido a ser rastreado, onde se encontra o usuário. Possui também a necessidade de lidar com oclusões, quando um braço ou mão se coloca na frente do outro.

Em jogos de computador, os personagens que representam seres humanos precisam ter a movimentação programada, de acordo com os movimentos que se espera que cada personagem possa realizar. Esses movimentos podem ser gravados usando o rastreamento como ferramenta de geração de dados sobre articulações e posições dos membros, para que a movimentação não tenha que ser programada de forma manual.

Também atividades que exijam manipulação de objetos em um Ambiente Virtual (AV), como em treinamentos de atividades de risco, montagem, manutenção, ou outros, pode se beneficiar de uma ferramenta de rastreamento que possibilite a captura com precisão de movimentos complexos e de maneira detalhada.

Se for estabelecido como requisito de tais sistemas de rastreamento uma fácil portabilidade, o uso de rastreadores mecânicos pode ser descartado. Se outro requisito for o

funcionamento em ambientes de configuração diversa, onde estarão presentes elementos passíveis de criar interferências, os rastreadores ultra-sônicos e eletromagnéticos também não poderão ser utilizados.

2.5 Tecnologias de rastreamento dos dispositivos utilizados no sistema

O Kinect e o Wii Remote são dispositivos atualmente disponíveis no mercado cujo uso se popularizou nos últimos anos e que utilizam tecnologias óticas e inerciais, respectivamente, para rastreamento de movimentos do corpo humano ou partes do corpo. Os dispositivos conhecidos como luvas de dados, para rastreamento da posição da mão e dedos, foram se tornando mais eficientes e com menor custo ao longo dos anos. A 5DT Data Glove Ultra é um exemplo de uso dessa tecnologia. Essas três tecnologias usadas em conjunto permitem configurar um sistema com as restrições descritas na seção anterior, de permitir uma alta mobilidade, o uso em ambientes diversos, e complementação das informações de rastreamento dos braços, antebraços e dedos. A seguir, aspectos importantes dessas tecnologias são apresentados.

2.5.1 Tecnologia ótica

Sistemas de rastreamento que utilizam visão computacional são sistemas que extraem informações de uma imagem (PITERI; RODRIGUES, 2011, p.113). A imagem é geralmente obtida de uma câmera que “enxerga” o ambiente, onde está o objeto a ser rastreado. O sistema então processa cada imagem obtida a fim de localizar o objeto na cena e determina sua posição (SHERMAN; CRAIG, 2003). O conjunto de posições ao longo do tempo pode ser traduzido em alguma informação relevante sobre o movimento efetuado pelo alvo.

As técnicas de visão computacional são muito utilizadas como método de entrada de dados na interação com computador em sistemas com dispositivos não convencionais. Isto

ocorre principalmente pelo fato de tal método poder ser utilizado sem a necessidade de dispositivos sofisticados.

Muitas aplicações de rastreamento fazem uso de câmeras estáticas digitais (DSC – *Digital Static Camera*), cujo custo tem decaído significativamente nos últimos anos. Outros métodos dependem do uso de dispositivos como luvas e sensores, que podem ser caros, ter problemas com relação a ergonomia, ou serem conectados ao usuário através de fios, restringindo os movimentos (SHERMAN; CRAIG, 2003).

2.5.1.1 Elementos Básicos

Os elementos básicos de um sistema de visão computacional são: câmera(s) ou outro dispositivo para captação de imagens, o cenário onde se encontra o objeto foco do sistema, e o hardware responsável pelo processamento.

O sistema pode possuir mais de uma câmera, que podem ser estáticas ou não, e de tecnologias variadas. Abrangem desde câmeras DSC simples, com as encontradas nos *notebooks* e celulares atuais, até câmeras profissionais, passando por câmeras de vigilância e com capturas de imagens em infravermelho (BURDEA; COIFFET, 2003).

Os dispositivos de captura de imagens possuem sensores de imagem sensíveis a diferentes frequências do espectro eletromagnético (Figura 2.7). As câmeras mais populares possuem sensores CCD (do inglês, *Charge-Coupled Devices*) que captam a luz visível ao olho humano (KOSCHAN; ABIDI, 2008). Alguns sistemas utilizam o infravermelho (IR, do inglês, *Infrared*) para a captura de imagens, conforme será discutido na próxima seção.

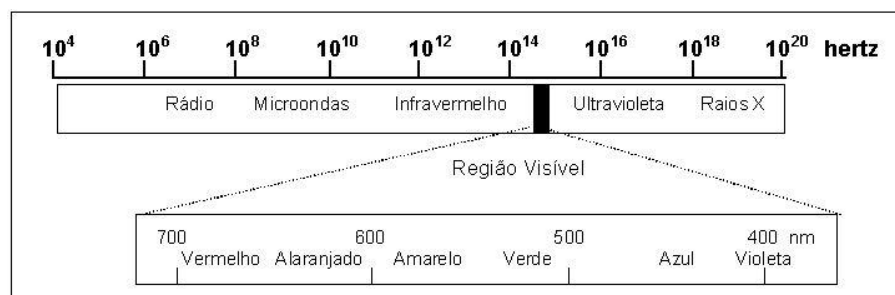


Figura 2.7 – Espectro eletromagnético. Fonte: (TRAINA, 2006).

Os sistemas que fazem uso de apenas uma câmera permitem a localização do alvo na imagem capturada retornando uma posição no plano bidimensional ou tridimensional, dependendo do algoritmo usado. Já os sistemas com duas câmeras, chamados de estereoscópicos, calculam de forma mais precisa as informações sobre a profundidade do alvo na cena, retornando a posição no espaço tridimensional (SHERMAN; CRAIG, 2003, p.81).

Tais sistemas se assemelham à visão binocular humana que, graças à diferença no ponto de vista proporcionado pela distância entre os olhos, permite a percepção de profundidade na cena. A configuração básica das câmeras em sistemas estereoscópicos consiste em posicioná-las a uma distância fixa da outra, além de realizar um processo de calibração, utilizando mais de um ponto de referência cuja localização é previamente conhecida (TORI; KIRNER; SISCOUTO, 2006; MAEDA et al., 2004).

O cenário pode consistir em um ambiente controlado, caso dos estúdios de gravação e laboratórios, ou ter uma composição de fundo e objetos aleatória, como em sistemas de vigilância de locais públicos. Além disso, os objetos podem possuir ou não marcadores óticos.

O sistema responsável pelo processamento da visão computacional é executado em um hardware apropriado à aplicação. Esse sistema realiza o armazenamento, temporário ou não, das imagens capturadas, no qual são aplicadas técnicas de processamento de imagens para distinguir os objetos de interesse na imagem, faz o rastreamento dos movimentos do objeto e interpreta o resultado dessas etapas.

2.5.1.2. Infravermelho

As técnicas de visão computacional para rastreamento de movimento se utilizam também da tecnologia infravermelho para captura de imagens. O uso desse espectro de banda eletromagnético é explorado por vários sistemas de visão computacional através da utilização de dispositivos que permitem a detecção da luz IR. Geralmente, tais sistemas fazem uso de uma câmera sensível à luz IR ou com um filtro IR, e emissores deste tipo de luz.

Jha (2000, p. 273) descreve algumas propriedades e características de sistemas de visão de máquina usando IR. Tais sistemas são compostos por um sensor para captar a imagem, um computador de alta velocidade que processa e interpreta a informação, e uma unidade de interface controlada por máquina que mostra o resultado.

São utilizados nas indústrias para controle automático da taxa de produção de um produto, monitoramento do controle de qualidade e integridade do processo, ou guiar a precisão dos movimentos da máquina envolvida no processo de produção. A intensidade da fonte de IR, a irradiância e a emissão do foco luminoso são alguns dos parâmetros de performance do sistema. O autor também esclarece sobre alguns fatores que interferem no desempenho de sistemas que utilizam tecnologia IR de modo geral (JHA, 2000, p. 31):

“Todos os sinais eletro-óticos experimentam absorção, reflexão, difusão e dispersão enquanto atravessam a atmosfera. [...] A transmissão de radiação IR na atmosfera [...] depende de absorção e dispersão, [...] que são fatores atenuantes do sinal [...] e importantes fatores no design de sensores IR. [...] A transmissão atmosférica é uma função complexa altamente dependente do comprimento de onda do laser IR, caminho de propagação, altitude, condições sazonais e variações na densidade e temperatura da atmosfera” (tradução nossa).

Para aplicações de rastreamento de movimentos, a tecnologia IR é usada para facilitar a detecção de alvos. Alguns sistemas utilizam o IR para fazer o rastreamento térmico de pessoas e animais (COLANTONIO et al., 2006). Tais sistemas possuem uma câmera capaz de captar a luz infravermelha, seja através de um filtro ou do próprio sensor de imagem usado na câmera. Já os sistemas focados em reconhecimento de gestos contêm marcadores para permitir maior precisão na localização e rastreamento.

Os marcadores podem ser ativos ou passivos (DORFMÜLLER; WIRTH, 1998). Os ativos são compostos de emissores de luz infravermelha, como *Light Emitting Diodes* (LEDs), que podem ser dispostos em locais fixos no ambiente enquanto a câmera permanece móvel (rastreamento videométrico), como é visto em (MAEDA et al., 2004), ou estarem presos a dispositivos usados pelo usuário, enquanto a câmera está em uma posição fixa (FOURSA, 2004; DORFMÜLLER; WIRTH, 1998). Os marcadores passivos são feitos de material que reflete o IR emitido por uma fonte externa presente na configuração do sistema (DORFMÜLLER-ULHAAS; SCHMALSTIEG, 2001). O uso deste último torna o sistema mais suscetível a interferências, já que podem ocorrer falsas reflexões. Ambos métodos estão sujeitos a interferência por luz de fundo no ambiente (STIEFMEIER et al., 2006).

2.5.2 Tecnologia inercial

“Os dispositivos inerciais modernos são estruturas no estado sólido que usam tecnologia de sistemas micro-eletromecânicos.” (BURDEA; COIFFET, 2003, p. 38, tradução nossa). Isto implica em dispositivos pequenos, leves e com baixo consumo de energia, que podem ser colocados em diferentes objetos ou partes do corpo a serem rastreadas. A Figura 2.8 apresenta um dispositivo inercial moderno disponível no mercado, composto por sensores inerciais.



Figura 2.8 – Dispositivo de rastreamento inercial comercializado pela Xsens, composto por giroscópio, acelerômetro e magnetômetro. Fonte: (XSSENS, 2011).

Este tipo de dispositivo tem sido bastante utilizado em conjunto com outras tecnologias em sistemas de rastreamento (PUSTKA; KLINKER, 2008). O tipo de informação captado por giroscópios e acelerômetros é complexo, e em alguns casos impossível de ser adquirido por outros meios.

O giroscópio contém um mecanismo interno composto por três unidades ortogonais entre si, capazes de medir os ângulos de inclinação do dispositivo nos três eixos cartesianos. Quando ocorre uma inclinação, o mecanismo é deslocado de sua posição inicial devido à inércia. Esse deslocamento é medido e transformado em informação sobre o movimento realizado, indicando a orientação do dispositivo em relação à posição inicial (ERİŞMIŞ, 2004).

Erişmiş (2004) explica o funcionamento do acelerômetro. Nele, uma estrutura interna ao mecanismo é presa a uma parte fixa através de uma mola. Quando há uma aceleração externa, a estrutura sofre um deslocamento da posição inicial de repouso. A amplitude do deslocamento é proporcional a aceleração. Essa amplitude é transformada em dados de aceleração do dispositivo.

2.5.3 Tecnologias usadas em luvas de dados

As luvas são dispositivos usados tanto para rastreamento (entrada passiva) quanto para interação (entrada ativa), pois a movimentação da mão e dos dedos pode servir para acionar comandos específicos de um sistema de RV.

Várias tecnologias podem ser usadas na elaboração das luvas. Existem luvas mecânicas, com dispositivos hápticos que se assemelham a um exoesqueleto que medem ou executam força e movimento nos dedos do usuário (Figura 2.9). Outra tecnologia utilizada é a de visão computacional, onde marcadores são dispostos nos pontos de interesse da luva, e uma câmera capta as imagens que o sistema irá processar para localizar tais marcadores (Figura 2.10). Existem ainda luvas que usam dispositivos de ultra-sons ou magnéticos para permitir seu rastreamento (STURMAN; ZELTZER, 1994; TORI; KIRNER; SISCOUTO, 2006).



Figura 2.9 – Luva Haptic Telexistence. Fonte: (SATO, 2007).



Figura 2.10 – Luva com marcadores óticos reflexivos. Fonte: (DORFMUELLER-UHAAS, K; SCHMALSTIEG, 2001).

Uma das tecnologias mais encontradas nas luvas que permitem ao sistema saber a flexão dos dedos é a de fibra ótica. Tori, Kirner e Siscouto (2006, p. 41) definem essas luvas: “Seu uso consiste em um fio de fibra ótica com junções. Quando a junta é movida, o cabo dobra-se reduzindo a passagem de luz por ele. Essas variações de luz são resumidas e transmitidas para o computador”. A Figura 2.11 apresenta o esquema dessa luva e a Figura 2.12 mostra a 5DT Data Glove Ultra, que utiliza fibra ótica.

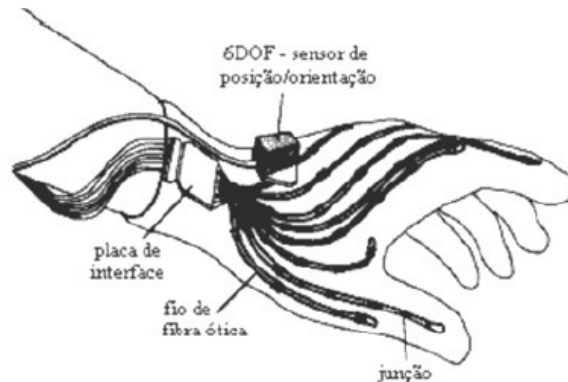


Figura 2.11 – Elementos de uma luva com fibra ótica. Fonte: (NETTO; MACHADO; OLIVEIRA, 2002).



Figura 2.12 – Luva da 5DT com sensores de fibra ótica. Fonte: (FIFTH DIMENSION TECHNOLOGIES, 2011).

As luvas de fibra ótica possuem as mesmas vantagens apresentadas pelos dispositivos inerciais. Os sensores de fibra ótica costumam ser leves, pequenos e não sofrem de interferências elétricas ou magnéticas.

A fibra ótica é feita de material translúcido e é composta por um núcleo, um revestimento e a cobertura da fibra (JHA, 2000), conforme ilustrado na Figura 2.13. O sensor com fibra ótica possui um emissor de luz em um dos extremos, geralmente um LED, e um receptor no outro.

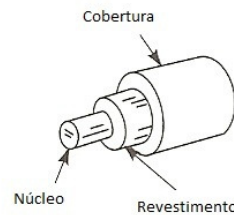


Figura 2.13 – Esquema de uma fibra ótica. Fonte: (JHA, 2000).

Os receptores acoplados à fibra ótica da luva medem a intensidade da luz recebida, permitindo indicar quão dobrados estão os dedos das mãos.

2.6 Considerações em relação ao rastreamento

O rastreamento de movimentos, independente da tecnologia, consiste basicamente em captar uma sequência de estados dos sensores e, baseado na diferença entre os estados em dois tempos distintos, indicar a posição inicial e a posição final do alvo que está sendo rastreado. O gesto pode ser reconhecido por meio da classificação das posições, ou conjunto de posições, inicial e final, efetuada com o auxílio de alguma técnica, como métodos de inteligência artificial, por exemplo.

As metodologias de classificação são independentes de tecnologia. Portanto, a maioria delas pode ser encontrada sendo aplicada em trabalhos com as mais diversas tecnologias.

2.6.1 Calibração

O rastreamento pode ser efetuado com um ou mais tipos de tecnologia. O uso de mais de um tipo melhora o desempenho do rastreamento, pois podem ser usados dispositivos com características complementares, segundo Sherman e Craig (2003). Os autores afirmam

também que, além da combinação de mais de uma tecnologia de rastreamento, este pode ser melhorado com a calibração do sistema ou com a análise de previsão.

A calibração consiste em colocar o objeto a ser rastreado em uma determinada posição conhecida, e informar ao sistema que o objeto está nesta posição, servindo de referência para o sistema. Em um sistema de visão computacional com uso de marcadores, por exemplo, a câmera é colocada em uma posição relativa aos marcadores, e o sistema é informado sobre a posição real da câmera e dos marcadores. A partir disso, a variação na localização da câmera (marcadores fixos) ou dos marcadores (câmera fixa) pode ser calculada com base na movimentação dos marcadores na cena.

Em sistemas que utilizam luvas de dados de fibra ótica, a calibração é feita para a mão fechada (dedos dobrados) e aberta, com o usuário indicando para o sistema qual estado do sensor corresponde a qual situação. Cada usuário deve fazer uma nova calibração cada vez que o sistema é iniciado (BURDEA; COIFFET, 2003).

Os sensores inerciais são calibrados indicando um estado inicial de repouso.

A calibração pode ser feita de forma automática por alguns sistemas. A tendência é automatizar esta etapa o máximo possível, principalmente em sistemas de visão computacional, onde ela pode se mostrar complexa de ser efetuada. Isto é necessário visto que muitos usuários de tais sistemas não são conhecedores dos métodos envolvidos (DORFMÜLLER-UHAAS; SCHAMLSTIEG, 2001; MAEDA et al., 2004).

2.6.2 Previsão

A análise de previsão é feita com base na trajetória do objeto em movimento. A partir dela, é possível fazer uma previsão de onde o objeto estará no instante seguinte, diminuindo a latência do sistema.

A previsão também pode ser útil quando ocorre oclusão em sistemas de visão computacional. Nesses casos, a posição do alvo pode ser estimada de acordo com uma previsão realizada com base nas posições anteriores na cena e trajetória. Esta técnica de

previsão é usada para acelerar o rastreamento em alguns casos. O rastreamento global é efetuado na imagem completa inicial. No *frame* seguinte, o rastreamento local é realizado na vizinhança da posição anterior, poupando trabalho computacional e, conseqüentemente, tempo (BAO; BINH; KHOA, 2009; FOURSA, 2004). Entretanto, por se tratar de uma previsão, nem sempre o resultado obtido é válido.

2.6.3 Parâmetros de desempenho

As tecnologias de rastreamento ótica, inercial e luva de dados com fibra ótica possuem quatro parâmetros de desempenho: precisão, ruído, latência e acúmulo de erros na localização (BURDEA; COIFFET, 2003).

O funcionamento da tecnologia ótica está sujeito a interferências na iluminação. O excesso de luz direcionada à lente da câmera ou a iluminação inadequada do ambiente podem dificultar ou até impossibilitar o sistema de localizar o alvo, segundo Burdea e Coiffet (2003). Quanto à taxa de atualização dos dados captados e a latência, os mesmos autores afirmam que tecnologias óticas, com o uso de visão computacional, têm melhor desempenho que os sistemas com tecnologia de ultra-sons, além de permitir uma área de atuação mais extensa.

Algumas vantagens dos dispositivos de inércia e da luva de dados com fibra ótica são: campo de atuação teoricamente ilimitado, não há problema de oclusão, baixa sensibilidade a ruído do ambiente, e constituem uma unidade autônoma, sem a necessidade de um segundo dispositivo para emissão ou captação de sinal. Os dispositivos de inércia, porém, possuem a desvantagem de acumular erros na leitura dos dados. Por isso requerem a recalibração manual periodicamente, ou, o que tem sido mais utilizado, a combinação de dispositivos de outras tecnologias em conjunto com os dispositivos inerciais (MAEDA et al., 2004; STIEFMEIER et al., 2006).

2.6.4 Rastreamento com visão computacional

O rastreamento de movimentos possui diferentes aspectos de acordo com a tecnologia empregada, assim como diferentes parâmetros. O rastreamento tridimensional baseado em sistemas de visão computacional é composto de duas etapas: processamento para extrair informação da imagem, e estimativa da posição (FUA; LEPETIT, 2007).

Em sistemas de visão computacional, características escolhidas de acordo com o sistema são procuradas a cada *frame*. Essas características podem ser marcadores óticos colocados com o objetivo de facilitar a localização e aumentar a precisão, características naturais do objeto, ou modelos pré-estabelecidos, como contornos, esqueletos, cores, texturas etc (FUA; LEPETIT, 2007).

Alguns fatores podem atrapalhar o rastreamento. Uma luz de fundo ou objeto na frente do alvo pode impedir a localização do mesmo. Dependendo da velocidade de movimentação dele e da capacidade de captura de *frames* por segundo da câmera, a imagem pode sair borrada e dificultar ou impossibilitar o posicionamento (FUA; LEPETIT, 2007).

Caso o alvo seja encontrado, a informação de sua posição é armazenada. A sequência de posições é analisada para tentar identificar o movimento efetuado usando um método de classificação escolhido (MOESLUND; GRANUM, 2001).

Wachter e Nagel (1999) realizam o rastreamento com o uso de um modelo em três dimensões (3D) do ser humano, procurando nas imagens captadas algo que se enquadre nas características do modelo, que possui DOFs estimados com o uso de filtros de Kalman.

Dorfmüller-Uhaas e Schmalstieg (2001) também utilizam modelos, porém como auxílio na identificação de movimento dos dedos, que são localizados com o uso de marcadores reflexivos. Filtros de Kalman também foram usados nesse trabalho.

Pan et al. (2010) e Feng et al. (2009) utilizam a cor da pele como característica a ser rastreada. O primeiro inicializa a localização usando o contorno da mão como característica a ser buscada no *frame*. O segundo usa uma variação do Filtro de Partículas para rastrear o movimento.

Estes trabalhos citados acima são alguns dos diversos existentes que testam diferentes configurações de sistemas de visão computacional. Cada um deles utilizando técnicas de localização, rastreamento e classificação variadas.

2.6.5 Rastreamento com sensores inerciais e luvas de dados

O rastreamento realizado com sensores inerciais ou com luvas de dados de fibra ótica é efetuado de maneira mais simples que na visão computacional. Isto se dá pelo fato de não haver a localização do alvo, apenas o rastreamento e classificação. O sistema precisa apenas ler e interpretar os dados dos movimentos obtidos pelos dispositivos.

Huyghe, Doutreloigne e Vanfleteren (2009) demonstram o uso de acelerômetros em conjunto com magnetômetro para rastreamento de orientação em 3D, com o uso de filtros de Kalman para tratamento de sinais.

Dong, Wu e Cheng (2007) configuraram um sistema que utiliza acelerômetros presos a partes do corpo para realizar a identificação do movimento por meio da combinação da leitura dos dados enviados pelos sensores. O mesmo princípio é usado por Yeoh et al. (2008) para determinar a flexão das juntas do usuário. Já Wu, Chang e Tseng (2011) tratam da minimização de distorções provocadas pela força da gravidade nesses tipos de sistemas.

Keir et al. (2007) e Avizzano et al. (2004) são exemplos de trabalhos com o acelerômetro para rastreamento da cabeça do usuário. Este tipo de aplicação também é apresentado por Satoh, Uchiyama e Yamamoto (2004) e Ogris et al. (2005) com o uso de giroscópios.

A luva de dados com fibra ótica transmite o sinal não tratado captado pelo sensor ao sistema computacional que irá utilizá-lo. O sistema então uniformiza esse sinal utilizando algum filtro, e o traduz em uma determinada configuração de posição dos dedos das mãos. A sequência de configurações pode ser entendida como um determinado gesto por meio do uso de redes neurais ou Modelos Ocultos de Markov (HMM – *Hidden Markov Models*), por exemplo (HOLDEN; OWENS; ROY, 1999).

Park, Kim e Hong (2008) apresentam um sistema de reconhecimento de gestos para interação com computador usando uma luva de dados com sensores de fibra ótica. Este é o principal tipo de aplicação para o rastreamento com luvas de dados.

2.7 Combinação de tecnologias

Como apresentado anteriormente, alguns sistemas fazem uso de uma combinação de tecnologias de rastreamento como forma de complementar os dados da entrada do usuário. Pontos fracos do rastreamento ótico, por exemplo, podem ser supridos pelo uso de um sensor em uma parte do objeto particularmente sensível à oclusão. Nos artigos referenciados neste trabalho, é possível observar que muitos sistemas têm seu foco no rastreamento de uma parte específica do corpo, como a cabeça ou as mãos. Entretanto, quando é necessário o rastreamento de mais de uma parte ou de um conjunto com nível de detalhamento diferente, caso das aplicações sobre línguas de sinais, o uso de duas ou mais tecnologias se mostra útil para obter os dados com a precisão desejada, como apresentado em alguns trabalhos citados a seguir.

Um exemplo é apresentado por Ogris et al. (2005) e por Stiefmeier et al. (2006), onde a incapacidade do rastreamento com ultra-sons em indicar a orientação do objeto foi sanada com o uso de sensores inerciais (Figura 2.14).

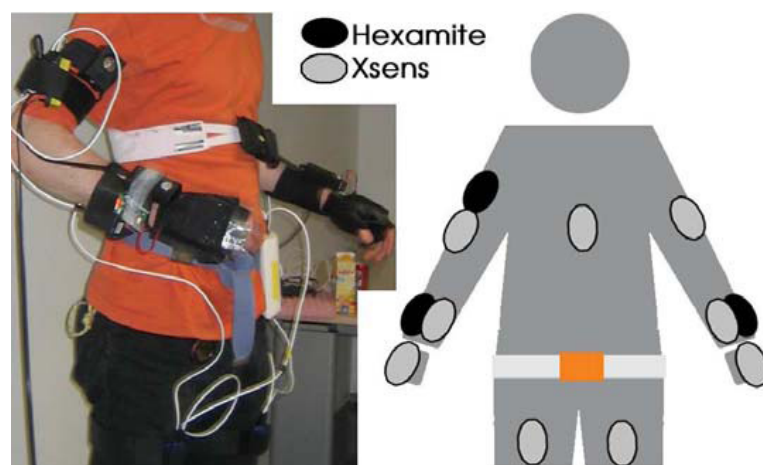


Figura 2.14 – Sistema de rastreamento usando sensores de ultra-sons (Hexamite) e inerciais (Xsens).

Fonte: (OGRIS et al., 2005).

Em Maeda et al. (2004), são utilizadas câmeras para localizar marcadores LED IR fixos no ambiente, e dispositivos inerciais. Esse conjunto é capaz de fornecer a inclinação, orientação da cabeça do usuário.

King et al. (2007) apresentam um sistema que utiliza uma luva composta por um sensor de fibra ótica que detecta a curvatura da palma da mão e acelerômetros para detectar a movimentação dos dedos.

O uso simultâneo de mais de uma tecnologia implica na necessidade de junção dos resultados. A classificação do movimento rastreado se torna mais complexa, pois deve ser efetuada levando em consideração duas entradas de dados em formatos diferentes. Isto pode ser feito com a fusão de classificadores, como exemplificado nos trabalhos de Ogris et al. (2005), Pustka e Klinker (2008) e Stiefmeier et al. (2006).

2.8 Considerações finais

A necessidade de maximizar a sensação de imersão e naturalidade de interação levou ao uso de métodos diferenciados de IHC em sistemas de RV. Várias tecnologias foram criadas para realizar a entrada de dados passiva, por meio do rastreamento de movimentos, em sistemas computacionais. A utilização dessas tecnologias, entretanto, não ficou restrita a aplicações em sistemas de RV.

Sistemas de rastreamento de gestos são possíveis graças a essas tecnologias. Porém, elas apresentam falhas para realizar, isoladamente, o rastreamento de movimentos de forma precisa para todo e qualquer tipo de sistema, com dispositivos de baixo custo e de uso intuitivo por leigos no assunto. A escolha de qual tipo de dispositivo usar deve ser feita com base no tipo de dado de entrada que o sistema precisa, assim como os parâmetros de desempenho que o mesmo exige.

O uso conjunto de mais de uma tecnologia se apresenta como uma boa solução para determinados sistemas, entre eles, o rastreamento de gestos. Alguns rastreamentos, de movimentos muito complexos e que exigem alto grau de precisão, podem ser facilitados e mais exatos se dois ou mais tipos de dispositivos forem usados.

O rastreamento com visão computacional permite um gasto baixo com dispositivo e possibilita tanto a localização quanto o rastreamento da direção do movimento. O uso de marcadores com IR facilita a localização, mas se torna um complicador por necessitar de hardware adicional.

Os dispositivos inerciais possuem boa portabilidade e fornecem dados sobre a orientação e velocidade do movimento sem apresentar problemas com interferências externas. Entretanto, não são capazes de fornecer a localização do objeto rastreado. Por isso, é comum o uso conjunto desses dispositivos com algum outro que consiga calcular a posição do objeto no plano ou no espaço.

As luvas de dados permitem saber a localização da mão e/ou posicionamento dos dedos com o uso de tecnologias diversas. Uma das mais usadas é a fibra ótica, que, assim como os dispositivos inerciais, não fornece a localização.

Cada grupo de tecnologia tem sido usado em dispositivos voltados para resolver determinados requisitos de aplicações. Na Tabela 2.1 é apresentado um resumo das vantagens e desvantagens das tecnologias discutidas neste Capítulo. Conforme a tecnologia evolui, os dispositivos têm se tornado mais acessíveis e as restrições ao uso tem diminuído.

Tabela 2.1 – Vantagens e desvantagens das tecnologias de rastreamento ótico, inercial e por luva de dados.

Tecnologia	Vantagens	Desvantagens
Ótica	• Dispositivos simples • Alta mobilidade • Baixo custo de equipamentos	• Sujeito à oclusão • Campo de atuação restrito
Inercial	• Dispositivos pequenos e leves • Baixo consumo de energia • Alta mobilidade	• Sujeito a acúmulo de erro na captura dos dados
De luvas de dados	• Alta mobilidade • Dispositivo simples de usar	• Custo pode ser alto • Dependendo da tecnologia, é sujeita a interferências

3 Dispositivos Comerciais de Rastreamento

Ao estudar as tecnologias de rastreamento e alguns produtos existentes que fazem uso de dispositivos com tal finalidade, foi possível observar que geralmente os sistemas possuem uma configuração complexa, com dispositivos desenvolvidos para serem usado por pessoas com conhecimento técnico, em ambientes controlados, na maioria das vezes com alto custo dos equipamentos, e priorizando o uso de um tipo de dispositivo apenas, em detrimento da precisão do rastreamento a ser efetuado.

A indústria de jogos eletrônicos foi a responsável por trazer ao usuário comum dispositivos de rastreamento fáceis de usar, de comprar, de preço acessível e combinando mais de uma tecnologia. O primeiro dispositivo cujo uso foi popularizado foi o Wii Remote, e mais recentemente, em 2010, o Kinect (ZHANG, 2012).

Esses dois dispositivos atendem os requisitos de baixo custo e facilidade de uso e de compra, além das vantagens apresentadas pelas tecnologias nos quais foram baseados. Se usados em conjunto também permitem a complementação dos dados e aumento de precisão no rastreamento corporal. Para completar o rastreamento, permitindo captar a configuração dos dedos, uma luva de dados de fibra ótica, como a 5DT Data Glove Ultra, pode ser utilizada.

Neste Capítulo esses três dispositivos são apresentados. O funcionamento de cada um será revisto, incluindo o tipo de sensor usado e a informação que fornece como entrada, assim como as ferramentas para desenvolvimento já existentes.

3.1 Kinect

O primeiro dispositivo selecionado utiliza uma das tecnologias mais empregadas em sistemas de rastreamento: a visão computacional. Além disso, tem sido alvo de vários pesquisadores em projetos com as mais diferentes finalidades (<http://www.kinecthacks.net/> e <http://www.kinecteducation.com/>). Esse dispositivo é o Kinect, lançado em 2010 pela

Microsoft para ser usado com seu console de jogos eletrônicos, o Xbox, como substituto ao controle tradicional.

Sua característica inovadora foi liberar o jogador da necessidade de segurar um controle e tornar o próprio corpo do jogador o controle. Juntando a isso o seu preço acessível, atualmente por volta de quinhentos reais, e a simplicidade de uso, o Kinect se difundiu rapidamente entre os jogadores e pesquisadores da computação. Estes últimos viram a oportunidade de utilizar uma ferramenta pronta e de baixo custo para auxiliar a interação por rastreamento em projetos variados.

O controle permite duas possibilidades de rastreamento: do corpo ou das mãos. O rastreamento do corpo permite que o dispositivo identifique e detecte movimentos do tronco, braços, pernas e cabeça. Já o rastreamento da mão permite a identificação de gestos feitos com uma das mãos para fins de interface com o computador ou console (XBOX, 2013).

3.1.1 Hardware

O Kinect utiliza uma combinação de duas tecnologias de sensores óticos para capturar dados da localização e movimentação do usuário. É dotado de uma câmera RGB, uma IR, e um emissor de IR (Figura 3.1), além de quatro microfones usados no reconhecimento de comandos por voz e um motor na base que direciona o controle para melhorar a visualização do usuário. A câmera RGB tem uma resolução de 640 x 480 pixels e é responsável pela captura da cena e pelo reconhecimento facial. A câmera IR captura a imagem formada pelo padrão de pontos criado pelo emissor IR. A partir dessa imagem o Kinect obtém as informações de profundidade da cena, pela triangulação da posição dos pontos na imagem e transformando a intensidade dos pontos refletidos em distância, o que permite o rastreamento em 3D (ZHANG, 2012).



Figura 3.1 – Sensores óticos do Kinect. Fonte: (KÜHN, 2011).

A conexão do dispositivo com o computador ou console se dá através de uma porta USB. O Kinect, no que tange ao rastreamento ótico, possui baixa latência do sinal e não atrapalha os movimentos do usuário com fios ou equipamentos pesados (ZHANG, 2012; MICROSOFT, 2013). Porém, seu alcance é limitado, e o usuário deve se posicionar a uma distância entre um metro e oitenta centímetros e três metros do equipamento para o rastreamento funcionar (XBOX, 2013), no caso do Kinect para o console Xbox. Já no Kinect para Windows, a distância mínima cai para quarenta centímetros (MICROSOFT, 2013).

3.1.2 Kinect SDK

A Microsoft lançou um *Software Development Kit* (SDK) proprietário para desenvolvedores programarem o Kinect para uso no computador. Mas bem antes disso a comunidade *open source* já havia lançado ferramentas de programação gratuitas. Uma das principais atualmente é a OpenNI (*Open Natural Interface*). A empresa PrimeSense, desenvolvedora do design que serviu de referência para o Kinect, lançou o middleware NITE e o *framework* OpenNI (KÜHN, 2011), que fornece uma API (*Application Programming Interface*) que pode ser usada tanto na comunicação com os sensores (baixo nível) quanto na comunicação com *middleware* para percepção de áudio e vídeo (alto nível) (PRIMESENSE, 2011).

O *middleware* NITE identifica o usuário e rastreia seus movimentos com o uso de algoritmos de visão computacional. Além disso, fornece uma API para implementar uma interface com o usuário baseada em gestos. Para funcionar com o Kinect, é utilizado um *driver* intermediário, o PrimeSense Sensor (Figura 3.2).

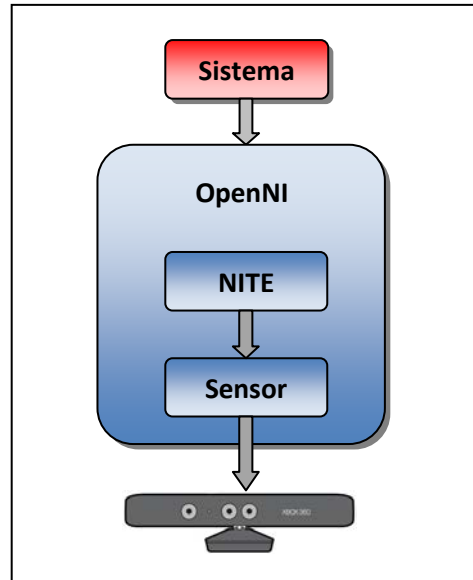


Figura 3.2 - Estrutura do OpenNI.

3.2 Wii Remote

O controle conhecido popularmente por Wiimote, lançado pela Nintendo em novembro de 2006 como parte do console Wii, apresentou um novo meio de interação em jogos eletrônicos, através de um controle sem fio cuja movimentação servia de entrada de dados juntamente com os tradicionais botões (LEE, 2008). O uso de sensores inerciais integrados a dispositivos eletrônicos começou então a se popularizar, com o lançamento de novos controles, celulares e smartphones dotados de tal tecnologia (ANTONIAC, 2008).

A presença de tais sensores no Wii Remote permite o rastreamento dos movimentos feitos com o controle. A partir de uma posição inicial de repouso, é possível detectar os movimentos feitos para os lados, para frente e para trás, e acima e abaixo (LEE, 2008). Isto é útil para complementar o rastreamento dos braços feito pelo Kinect, já que este falha na detecção da rotação dos antebraços e algumas vezes falha devido a oclusões na imagem.

Outras características favoráveis na escolha do Wii Remote para o sistema desenvolvido são o seu baixo custo, de aproximadamente duzentos reais, a portabilidade oferecida, e a simplicidade de uso, não requerendo conhecimentos específicos para sua manipulação, além de ser um dispositivo sem fio (NINTENDO, 2011).

3.2.1 Hardware

O Wii Remote é alimentado por duas pilhas tipo AA e se comunica com uma base, computador ou console, via tecnologia *Bluetooth*. É composto por doze botões, quatro LEDs, um alto falante para *feedback* sonoro, um motor de vibração para *feedback* tátil, uma porta de expansão, uma memória flash interna, uma câmera IR e um acelerômetro linear de três eixos. Este último possui uma sensibilidade de alcance de aproximadamente $3g^1$, oito bits por eixo e taxa de atualização de 100Hz (LEE, 2008). Para conectar o controle ao computador deve-se pressionar o botão “Sync”, localizado no compartimento de pilhas, ou os botões “1” e “2” simultaneamente para ativar o reconhecimento do Wii Remote via Bluetooth. O computador então detecta o dispositivo e realiza a sincronização. A Figura 3.3 mostra o controle e os eixos de referência sobre os quais funciona o acelerômetro.

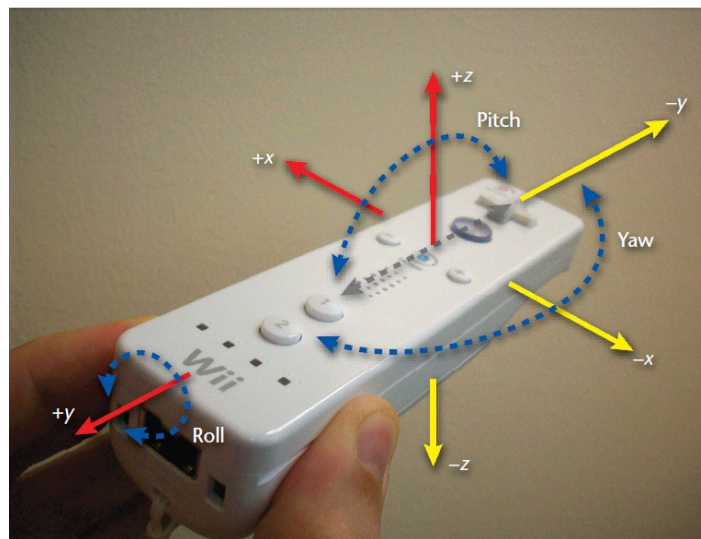


Figura 3.3 – Sistema de coordenada do Wii Remote. Fonte: (WINGRAVE, 2010).

3.2.2 Wii Remote SDK

A Nintendo disponibilizou um kit de desenvolvimento para o Wii, proprietário e com restrições em seu uso, mas desenvolvedores da comunidade de software livre criaram várias bibliotecas de software gratuitas para uso com o Wii Remote. Uma delas é a WiiuseJ

¹ g : Aceleração da gravidade em m/s^2 .

(DUCHE, 2009), uma API Java para conectar e usar o controle com o computador. Esta API pode ser usada tanto no sistema operacional Windows quanto no Linux e permite enviar comandos ao dispositivo, como acender os LEDs e fazê-lo vibrar, e recebe dados de eventos disparados pelo dispositivo, como o apertar de um botão ou uma informação do acelerômetro.

A empresa AiLive desenvolveu uma ferramenta proprietária, a LiveMove (AILIVE, 2009), que automatiza o reconhecimento de gestos feitos com o Wii Remote usando a técnica de aprendizado de máquina chamada aprendizagem por contexto. Os desenvolvedores de jogos para o Wii usam-na para tratar do rastreamento dos movimentos e reconhecimento dos gestos, podendo focar seus esforços de desenvolvimento no restante do software.

Um grupo de pesquisadores da Universidade de Oldenburg desenvolveu uma biblioteca baseada na linguagem Java de uso livre (sob a licença LGPL - *GNU Lesser General Public License*), chamada Wiigee, para reconhecer gestos previamente definidos, feitos com o uso do Wii Remote. A biblioteca permite que o desenvolvedor defina os gestos de interesse que usará na aplicação, treine os gestos com a ferramenta, que utiliza Modelos Ocultos de Markov, e obtenha a classificação automaticamente (SCHLÖMER et al., 2008).

3.3 Luva

O modelo de luva de dados escolhido para o trabalho foi o 5DT Data Glove Ultra, fabricado pela Fifth Dimension Technologies (FIFTH DIMENSION TECHNOLOGIES, 2011). Este é um dos modelos mais baratos atualmente no mercado, com um sensor de fibra ótica por dedo para medir a flexão. Além das vantagens da tecnologia utilizada na luva, já discutidas no Capítulo 2, esse modelo é de fácil uso, podendo ser conectado ao computador via cabo USB ou por meio de um *kit wireless* que pode ser adquirido em separado. Cada luva é vendida separadamente e pode ser adquirida direto com o fabricante, atualmente a um custo de novecentos e noventa e cinco dólares.

3.3.1 Hardware

A luva 5DT é uma luva de lycra e possui um sensor de fibra ótica por dedo. A resolução de flexão é de doze bits por sensor com uma taxa de amostragem mínima de 60 Hz (FIFTH DIMENSION TECHNOLOGIES, 2011). A Figura 3.4 ilustra o esquema de uma luva de mão direita, com as disposições dos sensores nos dedos identificados pelas letras de A a E. A Tabela 3.1 mostra, para cada um dos sensores, seu índice para o *driver* do dispositivo e a descrição da informação passada pelo índice.

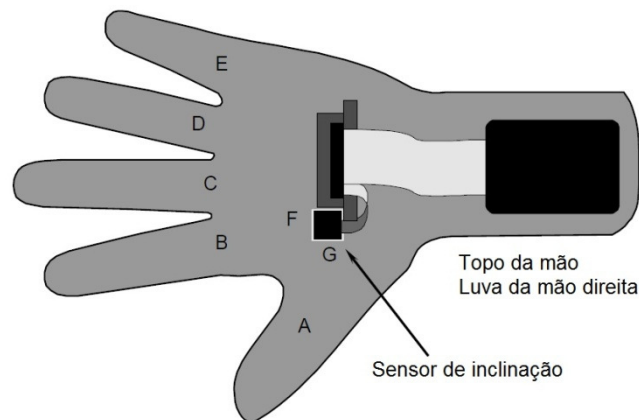


Figura 3.4 – Esquema da 5DT Data Glove Ultra. Fonte: (FIFTH DIMENSION TECHNOLOGIES, 2011).

Tabela 3.1 – Índices dos sensores para a 5DT Data Glove Ultra. Fonte: (FIFTH DIMENSION TECHNOLOGIES, 2011).

Sensor	Índice do <i>driver</i> do sensor	Descrição
A	0,1*	Flexão do polegar
B	3,4*	Flexão do dedo indicador
C	6,7*	Flexão do dedo médio
D	9,10*	Flexão do dedo anular
E	12,13*	Flexão do dedo mínimo
F	16*	Ângulo de inclinação com relação ao eixo horizontal perpendicular ao eixo longitudinal (sensor obsoleto)
G	17*	Ângulo de inclinação com relação ao eixo longitudinal (sensor obsoleto)

Espera-se que um dispositivo com tais características forneça dados com precisão. Ao ser inicializado, porém, não há parâmetros de referência para classificar o sinal de entrada. Esses parâmetros são dados na calibração, com a flexão de todos os dedos e posterior abertura

da mão. Por isso, o valor mínimo e máximo dos sinais de entrada varia de um usuário para outro. A Figura 3.5 mostra um exemplo dos sinais de entrada dos cinco sensores de uma luva. Cada faixa apresenta os dados relativos à flexão de um dedo ao longo do tempo, através do traçado de uma linha. A linha próxima à parte inferior da faixa indica que o dedo referente àquela faixa estava esticado. Quanto mais próxima do topo da faixa a linha estiver, mais flexionado estava o dedo.

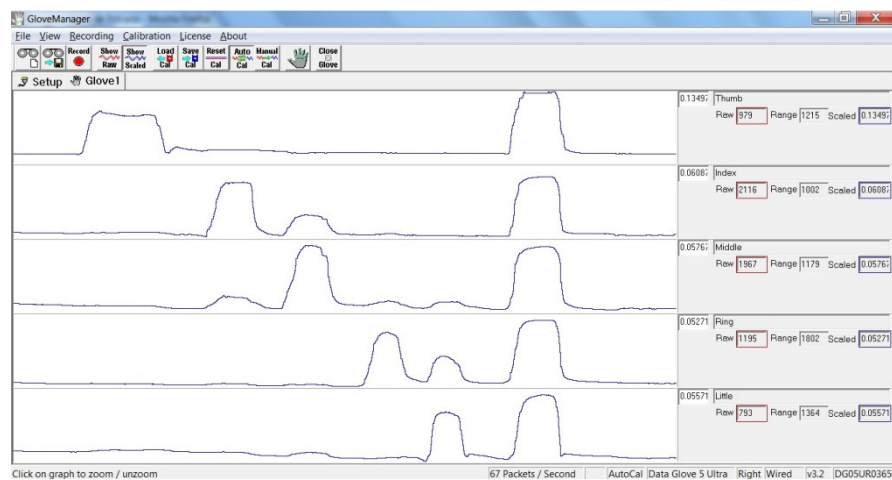


Figura 3.5 – Representação dos sinais capturados pela 5DT Data Glove Ultra. Fonte: (FIFTH DIMENSION TECHNOLOGIES, 2011).

3.3.2 5DT Data Glove Ultra SDK

Junto com a luva é fornecido um kit de desenvolvimento para várias plataformas em linguagens C/C++ e C#. Esse SDK permite automatizar tarefas de inicialização e desligamento das luvas, calibração e reconhecimento de gestos básico através de funções prontas. A Tabela 3.2 apresenta uma lista com os dados passados pela luva ao SDK para reconhecimento de gestos, e a ilustração com os gestos correspondentes é apresentada na Figura 3.6.

Tabela 3.2 – Esquema de definição de gesto implementado pelo SDK da 5DT Data Glove Ultra.
 Fonte: (FIFTH DIMENSION TECHNOLOGIES, 2011).

Nro. do Gesto	Flexão (0=flexionado, 1=estendido)				Descrição do gesto	Figura
0	0	0	0	0	Mão fechada	5.6.0
1	0	0	0	1	Dedo indicador estendido	5.6.1
2	0	0	1	0	Dedo médio estendido	5.6.2
3	0	0	1	1	Dois dedos estendidos	5.6.3
4	0	1	0	0	Dedo anular estendido	5.6.4
5	0	1	0	1	Anular e indicador estendidos	5.6.5
6	0	1	1	0	Anular e médio estendidos	5.6.6
7	0	1	1	1	Três dedos estendidos	5.6.7
8	1	0	0	0	Dedo mínimo estendido	5.6.8
9	1	0	0	1	Indicador e mínimo estendidos	5.6.9
10	1	0	1	0	Mínimo e médio estendidos	5.6.10
11	1	0	1	1	Apenas o anular flexionado	5.6.11
12	1	1	0	0	Anular e mínimo estendidos	5.6.12
13	1	1	0	1	Apenas o médio flexionado	5.6.13
14	1	1	1	0	Apenas o indicador flexionado	5.6.14
15	1	1	1	1	Mão aberta	5.6.15



Figura 3.6 – Ilustração dos gestos. Fonte: (FIFTH DIMENSION TECHNOLOGIES, 2011).

3.4 Considerações finais

Como exposto neste Capítulo, os dispositivos apresentados podem ser manipulados isoladamente por usuários comuns, isto é, que não tenham conhecimento técnico em rastreamento ou dispositivos de rastreamento, contanto que sejam utilizados com um software apropriado. Para permitir a utilização do Kinect, além do software ou jogo instalado, basta conectá-lo na porta USB do computador ou console, e colocá-lo em uma mesa ou suporte que permita o direcionamento de suas câmeras ao usuário, além de respeitar a distância mínima entre este e o aparelho. Para o Wii Remote funcionar no computador, o mesmo deve ter um dispositivo para comunicação Bluetooth, este deve estar ativado, e o software que fará uso do controle deve estar instalado. Basta então realizar a sincronização e movimentá-lo e/ou apertar seus botões de acordo com as instruções do software. Já a 5DT Data Glove Ultra precisa ser conectada uma em cada porta USB e vestida, posteriormente à instalação do software que utilizará seus dados.

Os três dispositivos apresentados possuem funções ou classes pré-programadas, permitindo a adaptação de seus usos em aplicações variadas. Espera-se que essa abstração do tratamento dos dados obtidos pelos sensores torne o desenvolvimento de aplicações mais rápido e em alto nível.

A luva 5DT capta com precisão a flexão dos dedos, mas não possui uma referência para o sinal de entrada e cada vez que é ligada ou troca de mãos precisa ser calibrada para que o sistema saiba quando o dedo daquele usuário em particular está flexionado e quando está esticado. Além disso, não faz o rastreamento da mão no espaço tridimensional.

O Wii Remote também possui problema com relação à falta de referência, no caso, no espaço tridimensional. Não é possível saber sua localização exata, mas a partir de uma posição inicial previamente estabelecida, o sistema pode calcular os movimentos realizados. O Kinect por sua vez fornece uma referência de localização do usuário no espaço, mas possui as limitações características de um dispositivo ótico. A Tabela 3.3 apresenta um comparativo entre os dispositivos discutidos neste Capítulo, classificando a mobilidade, precisão e facilidade de uso de cada um de acordo com as características tecnológicas apresentadas neste Capítulo e nos anteriores. Os três dispositivos são portáteis, por isso foram

classificados como tendo alta mobilidade. A precisão de cada um varia de acordo com a restrição imposta pela tecnologia utilizada (oclusões no caso de visão computacional, que podem ou não ocorrer; acúmulo de erro em dispositivos inerciais; e sensores de fibra ótica sendo o mais preciso entre os três). Podem ser instalados e manuseados por um usuário sem conhecimento técnico específico, portanto, são classificados como fáceis de usar. A luva possui o menor grau de liberdade entre os três, já que consegue captar movimento em apenas uma direção (eixo x) por sensor.

Tabela 3.3 – Comparativo entre os dispositivos.

	Mobilidade	Precisão	Programação	Facilidade de uso	Custo Aproximado (ago 2013)	Graus de Liberdade	Taxa de amostragem
Kinect	Alta	Média	Possui SDK	Fácil	US\$ 100,00	6 (x, y, z)	30 FPS
Wii Remote	Alta	Baixa	Possui SDK	Fácil	US\$ 30,00	6 (x, y, z)	100 Hz
5DT Data Glove Ultra	Alta	Alta	Possui SDK	Fácil	US\$ 995,00	1 (x)	60 Hz

4 Sistema proposto e desenvolvido

O Kinect possui baixa latência (MICROSOFT, 2013) e não atrapalha os movimentos do usuário com fios ou equipamentos pesados. Porém, no rastreamento de gestos complexos, como os usados para comunicação humana, ele não é suficiente, pois ainda não é preciso o bastante e não consegue fornecer toda a informação necessária, ou seja, os dados dos membros superiores e dos dedos ao mesmo tempo. O dispositivo não se comporta adequadamente ao rastrear objetos em escalas diferentes. Ao rastrear o corpo, o Kinect não tem precisão para rastrear as mãos detalhadamente, e quando rastreia as mãos não detecta o corpo (MICROSOFT, 2013). Além disso, a rotação dos antebraços não é captada pelos algoritmos de visão computacional, e problemas de oclusão podem ocorrer com frequência neste tipo de aplicação.

A Figura 4.1 mostra exemplos de rastreamentos incorretos do Kinect quando um usuário realiza gestos que acarretam oclusão. Em (a), a localização de um braço foi perdida quando este ficou oculto pelo outro braço. O mesmo problema ocorreu em (b), apesar de apenas uma pequena parte do antebraço e da mão estarem ocultos. Houve também a inversão na localização de um braço, mesmo problema ocorrido em (c) com os dois braços. Em (d) foram invertidos não apenas a direção (os braços abertos foram indicados como estando cruzados na frente do corpo), como os ângulos nas juntas dos cotovelos (AGOSTINHO; BREGA; DIAS, 2011). Portanto, para classificar corretamente e reproduzir os gestos rastreados, é preciso complementar as informações.

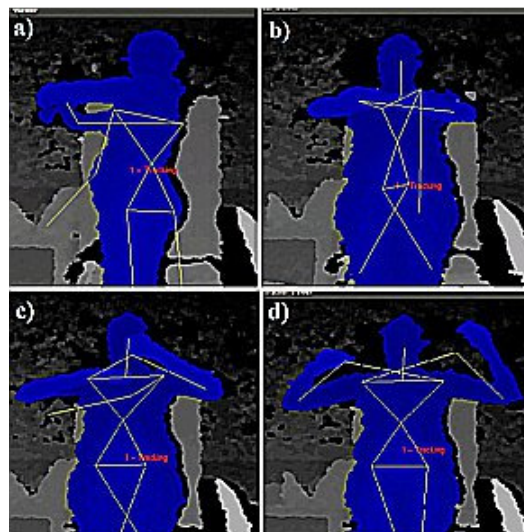


Figura 4.1 – Exemplos de erro no rastreamento feito pelo Kinect devido a oclusões.

Durante a realização desta pesquisa foram observados alguns projetos utilizando com sucesso a combinação de dispositivos para aperfeiçoar o rastreamento em atividades específicas, como mencionado no Capítulo 2. Mas durante o levantamento bibliográfico, foi notado que essa combinação ainda foi pouco analisada para aplicações de rastreamento de gestos. Além disso, não foi encontrado um sistema de baixo custo, com ferramentas simples de usar e que faça a captura de movimentos de braços, antebraços e dedos, simultaneamente.

Foram observados dois usos para os dados do movimento rastreado nos trabalhos pesquisados: a gravação dos dados para uso posterior, ou a reprodução do movimento em alguma ferramenta de vídeo.

Dada a natureza tridimensional dos gestos como os das línguas de sinais, por exemplo, é necessária uma maneira de reproduzi-los de forma que permita uma visualização eficiente dos movimentos captados no rastreamento. Portanto, é interessante a criação de um sistema que represente de forma tridimensional o sinal rastreado e, para melhor visualização, deve ser associado com formas humanas. Neste caso então, a utilização de um Humano Virtual em um ambiente 3D se torna interessante na sua representação.

O sistema foi projetado e implementado com o uso de um Kinect, dois Wii Remote (cada um afixado em um antebraço do usuário), e um par de luvas, conforme ilustrado na Figura 4.2.

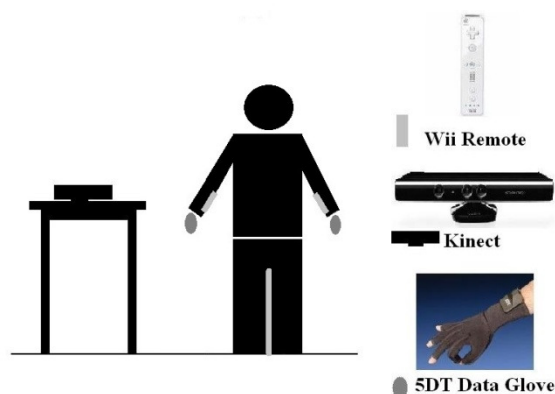


Figura 4.2 – Proposta de combinação para rastreamento de gestos.

Foram implementados cinco módulos no sistema. Um para calibração e leitura de cada dispositivo de forma independente, um módulo principal para iniciar todo o software e

realizar o controle dos demais e a sincronização, e um módulo para reprodução dos movimentos em tempo real. Este último foi adaptado de um protótipo de tradução de gestos em Língua Brasileira de Sinais (LIBRAS) que faz uso de um HV (BREGA, 2010) como recurso de visualização dos movimentos rastreados.

O sistema operacional utilizado para o trabalho foi o Linux com a distribuição Ubuntu, já que o custo é fator a ser considerado e se trata de um sistema operacional distribuído gratuitamente sob a licença LGPL.

Foram utilizados os pacotes de desenvolvimento gratuitos existentes para cada dispositivo, sendo os módulos e o protótipo do HV implementados na linguagem Java. A biblioteca da luva foi integrada ao sistema por meio de *Java Natural Interface* (JNI). O Wii Remote foi programado diretamente no Java, por possuir biblioteca nesta linguagem, assim como o Kinect, cuja biblioteca *open source* escolhida é adaptada para uso em programas Java. Cada módulo do sistema, esquematizado na Figura 4.3, está descrito na Seção 4.1.

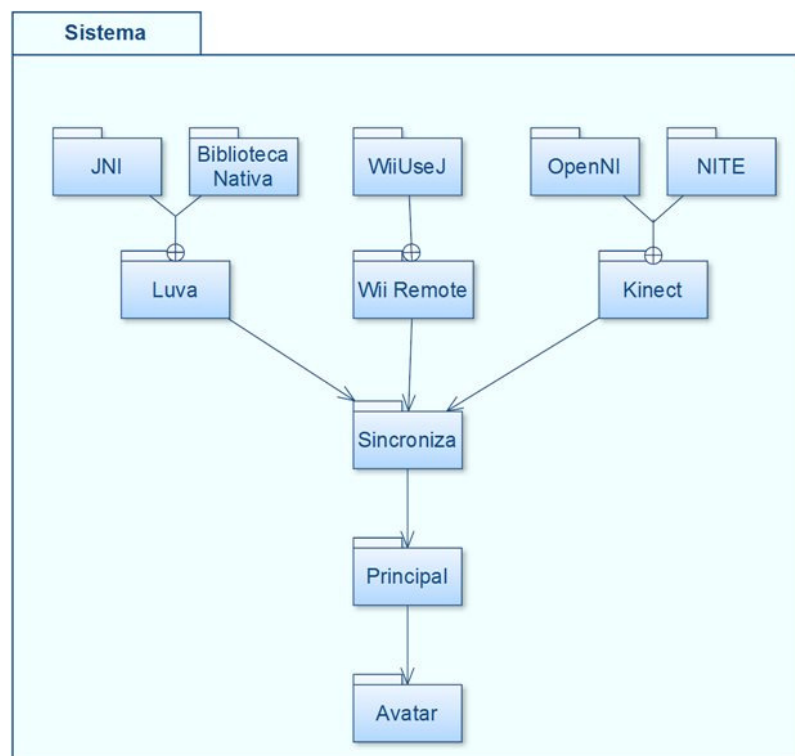


Figura 4.3 – Diagrama de módulos do Sistema.

Como o protótipo do HV empregado estava em um arquivo no formato de um projeto da IDE (*Integrated Development Environment*) Netbeans, esta foi escolhida para a codificação do sistema.

4.1 Módulos

Cada módulo correspondente aos dispositivos foi criado como um ou mais pacotes Java independentes que fazem uso de funcionalidades dos módulos principal e de movimentação do HV. Isto foi feito para permitir a inicialização, rastreamento e o teste individual de cada dispositivo antes de sua sincronização.

4.1.1. 5DT Data Glove Ultra

Um par de luvas 5DT Data Glove Ultra foi utilizado no sistema, conectadas ao computador através de um cabo USB para cada luva. Porém, existe a possibilidade de utilizar um “*kit wireless*” para conexão das luvas, que é vendido separadamente, o que permite a movimentação do usuário ocorrer livremente devido à ausência de cabos (FIFTH DIMENSION TECHNOLOGIES, 2011).

4.1.1.1 Funções e desenvolvimento

As principais funções de controles foram disponibilizadas pelo fabricante nas linguagens C/C++ e C# para o uso e controle das luvas. As funções responsáveis pela inicialização das luvas, identificação das mãos (se direita ou esquerda), fechamento e leitura dos dados foram incorporadas na biblioteca DataGLove5DT, escrita usando a linguagem C++, que realiza as chamadas para as funções do fabricante, passando como parâmetro as portas USB para inicialização, e a gravação dos dados lidos em um vetor, conforme esquema representado na Figura 4.4.

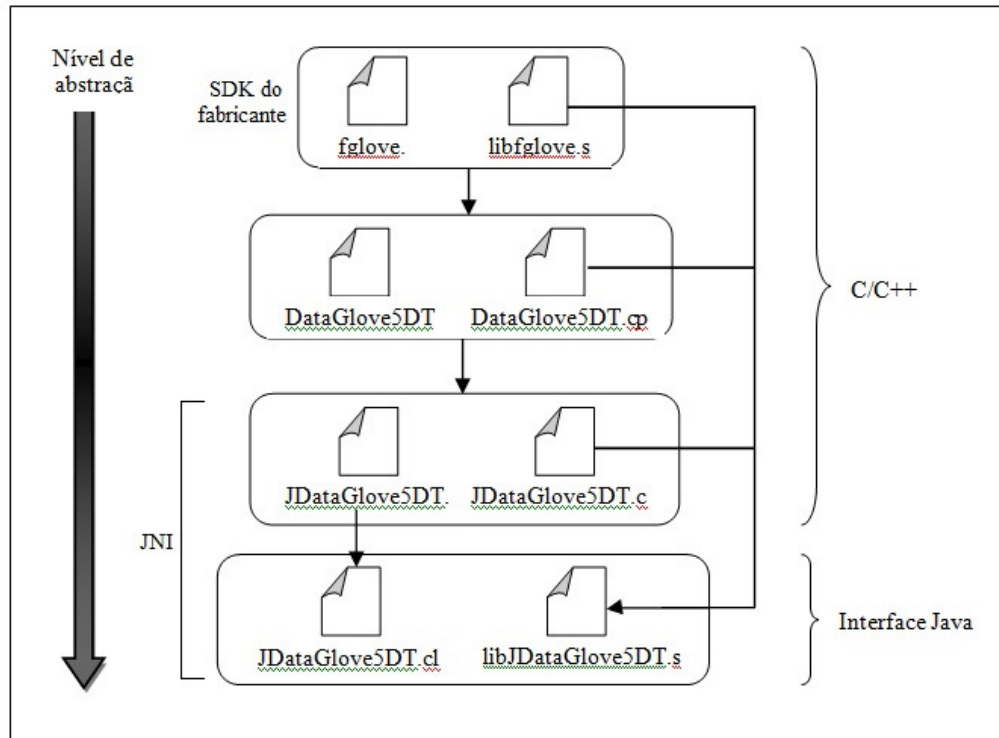


Figura 4.4 – Organização dos arquivos de acesso às luvas.

Para permitir a integração dessa biblioteca C++ criada, no sistema escrito em Java, foi empregado o *Java Native Interface*. O JNI é uma ferramenta que permite que um código executando em uma máquina virtual Java possa instanciar e ser instanciado por aplicações nativas ou bibliotecas em outras linguagens de programação (SUN, 2012).

Uma classe Java foi escrita (*JDataGlove5DT*) com os métodos nativos a serem chamados, e com o JNI foi gerado o arquivo de interface desta com o código C++. A partir dos arquivos C++ criados gerou-se a biblioteca `libJDataGlove5DT.so`, que em conjunto com a classe *JDataGlove5DT* permite que a aplicação Java acesse as luvas.

O método de leitura dos dados utilizado foi o *fdGetSensorScaledAll*, que fornece a implementação de uma calibração automática e contínua. Um valor máximo e mínimo do sinal de cada sensor é definido no início. Conforme os sinais dos sensores vão sendo recebidos, os valores absolutos vão sendo comparados com os valores máximo e mínimo atuais. Se o valor recebido for maior que o máximo ou menor que o mínimo, o valor ultrapassado é substituído pelo atual.

Um exemplo da interação dos códigos C++ com o Java é na conexão das luvas. Para iniciar esta, no pacote *Luva5DT* do sistema, a classe *Luvas* possui o seguinte método apresentado no trecho de Código 4.1.

Código 4.1 – Método para conectar luvas presente no sistema.

```
private void ConectarLuvas() {

    int retorno;
    try {
        retorno=Gloves.openGlove();
    }
    catch (Exception e) {
    }

    ...
}
```

Sendo que *Gloves* é um objeto da classe *JDataGlove5DT*. Esta, possui a assinatura do método *openGlove* apresentada no Código 4.2, que referencia para a chamada de método com JNI presente na *libJDataGlove5DT.so*. Esta chamada está descrita no arquivo *JDataGlove5DT.cpp* (Código 4.3).

Código 4.2 – Carregamento da biblioteca da luva e assinatura do método de conexão das luvas.

```
public class JDataGlove5DT {

    static {
        System.loadLibrary( "JDataGlove5DT" );
    }

    public native int    openGlove();
    ...
}
```

Código 4.3 – Chamada de método pelo JNI em *JDataGlove5DT.cpp*.

```
JNIEXPORT jint JNICALL Java_JDataGlove5DT_openGlove(JNIEnv *,
jobject) {

    jint retorno;

    retorno=dataGlove5DT->openGlove();

    return retorno;

}
```

O objeto *dataGlove5DT* é uma instância da classe *DataGlove5DT*, que por sua vez possui o método *openGlove* que implementa a inicialização das luvas, retornando um número inteiro de acordo com o sucesso ou não da operação, conforme trecho de Código 4.4.

Código 4.4 – Método de conexão das luvas em *DataGlove5DT.cpp*.

```
int DataGlove5DT::openGlove() {  
  
    szPortA = portaA;  
    szPortB = portaB;  
  
    pGloveA = fdOpen(szPortA);  
    pGloveB = fdOpen(szPortB);  
  
    if (pGloveA == NULL)  
    {  
        return 1;  
    }  
    if (pGloveB == NULL)  
    {  
        return 2;  
    }  
    return 0;  
  
    ...  
}
```

A função *fdOpen* é implementada pela biblioteca da luva fornecida pelo fabricante, e retorna um ponteiro para a luva conectada na porta USB passada como parâmetro na chamada da função.

Foi gerado um arquivo compactado .jar com a classe *JDataGlove5DT* e as bibliotecas da luva, e adicionado ao projeto Java. Este trata apenas com a classe adicionada, ocultando toda a implementação em código nativo para manipular a luva.

A Figura 4.5 mostra a sequência de uso das funções no sistema. Primeiramente, uma busca é feita para localizar as luvas nas portas USB do equipamento, para então inicializá-las. Os valores iniciais de calibração são definidos. Os valores relativos dos sensores são obtidos e transformados em ângulos. Após a finalização do uso, uma função específica finaliza as luvas e libera as portas USBs.

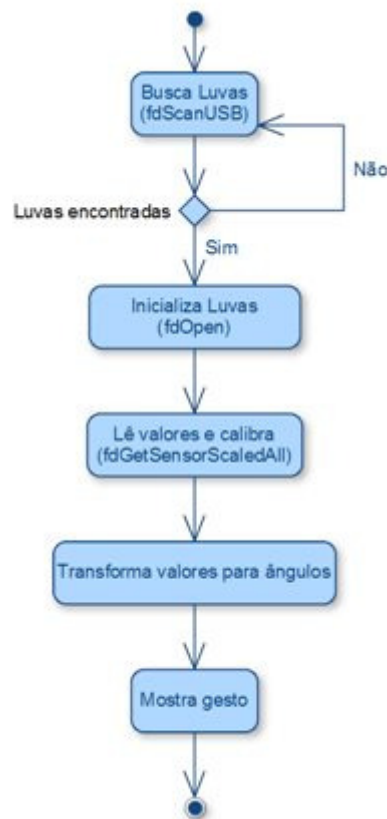


Figura 4.5 – Esquema de funcionamento do módulo da 5DT Data Glove.

Os valores relativos lidos de cada sensor das luvas podem variar de 0.0 a 1.0, sendo o primeiro valor equivalente ao dedo totalmente esticado e o segundo, flexionado. A conversão desses valores para ângulos foi feita usando o Teorema de Tales (HARUNA, 2000). Foi estabelecido um ângulo α imposto como limite da flexão dos dedos no HV, sendo α igual a 90° para o polegar, e 180° para os demais dedos. A seguinte equação descreve o meio de obtenção do ângulo β para uso no HV, a partir do valor de sensor x e do limite α :

$$\beta = \alpha * x$$

4.1.1.2 Interface

Uma interface gráfica foi criada para acessar os controles das luvas e exibir mensagens de acordo com o resultado da execução (Figura 4.6). O botão “Conectar luvas” faz a conexão com as luvas que estiverem conectadas ao computador via USB e identifica cada uma como sendo direita ou esquerda. Se bem sucedido, ou a conexão falhar, ou alguma luva não for encontrada, uma mensagem correspondente é exibida no rótulo abaixo do botão. O

botão “Ativar” inicia uma *thread* que realiza repetidamente a leitura dos sensores, normalização dos valores e passagem destes para serem reproduzidos no HV. O botão “Desconectar” interrompe essa *thread* e encerra a conexão das luvas.



Figura 4.6 – Interface gráfica para controle das luvas.

4.1.1.3 Exibição dos resultados

Inicialmente, os valores lidos das luvas foram exibidos no terminal do Netbeans, para verificar se a troca de dados entre o programa Java e as bibliotecas C++ estavam ocorrendo, e se a *thread* funcionava corretamente. Posteriormente, com a integração com o módulo do HV, descrito na seção 4.1.4, o movimento pôde ser visualizado em um HV em tempo real.

4.1.2 Wii Remote

Para que o movimento de rotação dos antebraços fosse captado, um Wii Remote foi afixado em cada um desses membros. Os controles foram posicionados próximo aos pulsos e no lado inferior, oposto ao local de conexão do cabo da luva, para que este não atrapalhasse o posicionamento do controle, paralelo ao antebraço (Figura 4.7).



Figura 4.7 – Posicionamento do Wii Remote.

4.1.2.1 Classes e desenvolvimento

Um módulo na linguagem Java foi desenvolvido com base no trabalho de Marca (2011), utilizando a API WiiuseJ, que implementou um rastreador de braços e antebraços utilizando quatro Wii Remotes. Neste trabalho, o módulo é composto de um pacote contendo as classes de interface com o hardware e controle do mesmo.

A classe *ControlesWiimote* implementa a localização e conexão dos controles via Bluetooth, a ativação dos mesmos para que a leitura dos sensores seja iniciada, e a desconexão. Após a conexão, cada controle é associado a uma instância da classe *MembroSuperior* ao se pressionar a tecla “A” do mesmo, atividade esta realizada pela classe *WiimoteEventos*. A classe *MembroSuperior* recebe os dados dos acelerômetros, passados pelo método *onMotionSensingEvent*, sendo o valor absoluto de *roll* o ângulo da rotação do mesmo. Esses valores são convertidos de acordo com o sistema de eixos do HV, para então serem usados na movimentação deste. A Figura 4.8 apresenta o diagrama do módulo do sistema que trata do rastreamento com o Wii Remote.



Figura 4.8 – Diagrama de atividades do módulo de rastreamento pelo Wii Remote.

Os dados do sensor são convertidos diretamente para valores em radianos, sem a necessidade de outro cálculo adicional como no caso das luvas, pois são fornecidos em graus. O ângulo então é passado para a classe que realiza a movimentação do HV.

4.1.2.2 Interface

A interface do módulo do Wii Remote, apresentada na Figura 4.9 permite a interação com botões e *feedback* visual conforme descrito a seguir.



Figura 4.9 – Interface gráfica para interação com os controles Wii Remote.

Ao pressionar os botões “1” e “2” do Wii Remote simultaneamente, ativa-se o reconhecimento do controle. Com o modo de reconhecimento ligado, ao pressionar o botão “Conectar wiimotes” do sistema, o controle pode ser localizado via Bluetooth e conectado ao computador. Quando a conexão ocorre, a identificação de qual Wii Remote está conectado a qual membro, é feita seguindo instruções que aparecem na tela, pressionando o botão “A” do controle correspondente ao membro descrito no topo da tela e em destaque no desenho apresentado ao lado esquerdo da interface. A barra abaixo do nome de cada membro, passa então a mostrar o nível de bateria do controle associado ao membro.

Após a identificação, o botão “Ativar” inicia a leitura dos sensores e envio dos dados ao HV. É possível ainda voltar o mesmo para a posição inicial através do botão “Resetar” e controlar a sensibilidade com que os movimentos são reproduzidos por meio da barra de controle “ROLL” na parte inferior da interface.

4.1.2.3 Exibição dos resultados

Inicialmente, um *feedback* de texto no terminal do Netbeans permitiu a visualização dos ângulos obtidos. A partir da integração com o módulo do HV, foi possível a visualização da rotação dos antebraços de forma tridimensional e independente do rastreamento dos braços.

4.1.3 Kinect

Um Kinect foi utilizado para rastrear os movimentos dos braços e antebraços do usuário. O *framework* OpenNI foi usado, permitindo a integração do dispositivo ao sistema, em conjunto com o middleware de visão computacional 3D NITE. Sua API possui várias funções e objetos pré-definidos que generalizam a troca de dados da aplicação com o sensor, uniformizando a programação (PRIMESENSE, 2011). O NITE permite a identificação do usuário na cena e o rastreamento de juntas.

4.1.3.1 Classes e desenvolvimento

O módulo do Kinect, o pacote MMSKinect, possui três classes: uma para realizar a interface com o usuário, uma com o dispositivo e uma com o HV. A primeira classe, *Kinect*, cria um objeto que representa o dispositivo a ser manipulado.

A segunda, *KinectUserTracker*, é responsável pela localização do usuário na cena e identificação de poses, que permite detectar a pose de calibração, conhecida como “Psi”, a partir de onde um esqueleto do usuário é montado e passa a ter suas juntas rastreadas. Essa classe executa uma *thread* que lê continuamente a orientação das juntas dos ombros e cotovelos, extrai os valores dos ângulos correspondentes, e os envia para a classe de interface com o HV, contanto que exista um usuário calibrado sendo rastreado.

A orientação das juntas é fornecida na forma de uma matriz de rotação 3 x 3, representando a rotação entre as coordenadas locais da junta e as coordenadas globais. Cada coluna é um vetor indicando a direção da junta em um dos eixos de coordenada global (PRIMESENSE, 2011). Os ângulos das juntas foram obtidos por meio da multiplicação das matrizes de rotação de acordo com o explicitado em (EBERLY, 2011), resultando no Código 4.5.

Código 4.5 – Método para obter os ângulos das juntas a partir da matriz de rotação.

```
private void ExtraiAngulos(Matrix3d pMatrix3d) {
    if (pMatrix3d.getM02() < +1)
    {
        if (pMatrix3d.getM02() > -1)
        {
            AngY = Math.asin(pMatrix3d.getM02());
            AngX = Math.atan2(-(pMatrix3d.getM12()), pMatrix3d.getM22());
            AngZ = Math.atan2(-(pMatrix3d.getM01()), pMatrix3d.getM00());
        }
        else
        {
            AngY = -Math.PI/2;
            AngX = -(Math.atan2(pMatrix3d.getM10(), pMatrix3d.getM11()));
            AngZ = 0;
        }
    }
    else
    {
        AngY = +Math.PI/2;
        AngX = Math.atan2(pMatrix3d.getM10(), pMatrix3d.getM11());
        AngZ = 0;
    }
}
```

A terceira classe do pacote, *MovimentaBracos*, recebe os ângulos e os envia para serem reproduzidos no HV quando necessário. Na Figura 4.10 está representado o diagrama do módulo do Kinect.

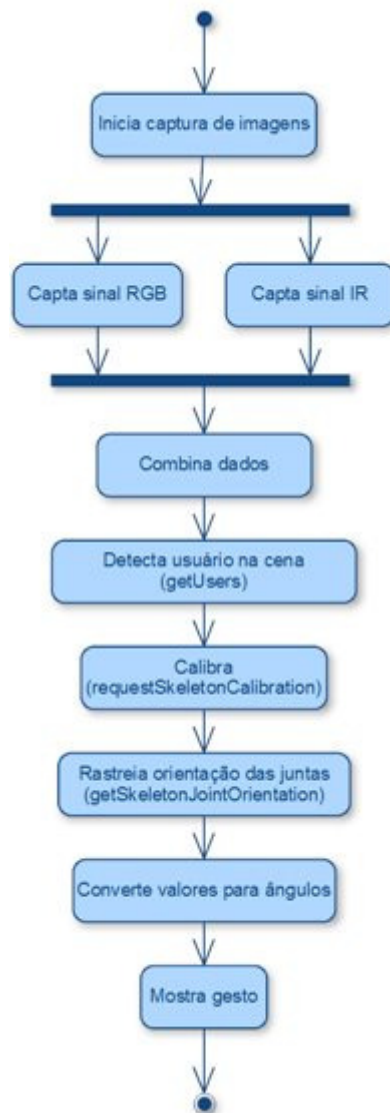


Figura 4.10 - Diagrama do módulo de rastreamento pelo Kinect.

4.1.3.2 Interface

A interface gráfica com o usuário para o módulo do Kinect é apresentada na Figura 4.11. O botão “Conectar Kinect” liga o dispositivo e inicia o rastreamento pelas câmeras. Abaixo deste botão são exibidas mensagens indicando quando um usuário é detectado na cena ou perdido, se a pose de calibração exibida na imagem ao lado esquerdo da interface é feita pelo usuário e reconhecida pelo sistema, quando a calibração ocorre e a indicação de que o rastreamento está em andamento.



Figura 4.11 – Interface do módulo do Kinect.

O botão “Desativar” interrompe a *thread* que realiza a leitura das juntas e o “Resetar” retorna o HV para a posição inicial.

4.1.3.3 Exibição do resultado

Um esqueleto com as principais juntas do usuário rastreado é formado, como ilustrado na Figura 4.1 anteriormente. Essa imagem formada em conjunto com os ângulos extraídos exibidos no terminal do Netbeans foram usados como *feedback* inicial no módulo de rastreamento do Kinect. Posteriormente à integração com o módulo do HV, os movimentos dos braços e antebraços puderam ser reproduzidos no HV.

4.1.4 Protótipo de Humano Virtual

A informação obtida pelo rastreamento pode ser repassada a um modelo humanoide para exibir a reprodução do movimento em um dispositivo de saída. Em sistemas

de ensino e treinamento, por exemplo, é interessante que o usuário possa visualizar em 3D os movimentos realizados, e os movimentos armazenados no sistema que sirvam de referência ao usuário, para que os dados relativos aos três eixos do espaço sejam reproduzidos e nenhuma informação se perca ao mostrar o movimento em apenas duas dimensões. Essa visualização 3D é facilitada com o uso de HV.

Os HV podem ser definidos como modelos de humanos que obedecem alguns requisitos, como aparência, autonomia e movimentação, por exemplo, e podem ser classificados como Agentes Virtuais ou Avatares. Este último consiste na representação gráfica de uma pessoa real, sendo, portanto, a forma adequada para o usuário simular os próprios movimentos (BREGA, 2010).

Os HV são usados em diferentes atividades e por motivos diversos, além do ensino e treinamento citado anteriormente: simulações de atividades de risco, poupando o usuário da exposição desnecessária ao risco real; testes ergométricos em produtos em fase de projeto, eliminando a necessidade de construção de protótipos físicos para testes preliminares; representação humana em jogos; etc (BREGA, 2010).

A modelagem de um HV deve atender alguns requisitos (MODESTO, 2005 *apud* BREGA, 2010): aparência, funcionalidade, tempo de produção, autonomia, individualidade, atributos físicos e biomecânicos, movimentação e inteligência.

A modelagem do corpo pode ser feita usando uma figura do tipo “palito” (representando o esqueleto e juntas do corpo), usando primitivas volumétricas simples (com figuras geométricas representando os membros), com modelos multicamadas (camadas representando os ossos, músculos e tecido adiposo) ou modelos corretos anatomicamente (com dados da medicina). Já a obtenção de uma forma humana a ser reproduzida pode ser obtida por reconstrução em três dimensões, baseada em foto ou vídeo (FUA et al., 2004).

Um sistema de ensino e treinamento pode apresentar, com o uso de um humano virtual, um movimento previamente armazenado. Isto pode ser usado pelo usuário para observar o movimento e tentar repeti-lo. No sistema desenvolvido, o Avatar fornece um meio de o usuário observar o movimento rastreado em tempo real.

Para isso, um módulo foi desenvolvido utilizando como base no protótipo contendo um Ambiente Virtual com um Avatar, implementados com o uso de Java 3D (SUN, 2013), descrito em (BREGA, 2010). Este Avatar foi desenvolvido seguindo o padrão H-anim

(H-ANIM, 2002), permitindo portanto, sua integração com sistemas que suportem esse formato. Da mesma forma, o sistema desenvolvido permite a troca do HV por outro que esteja neste mesmo padrão.

Este módulo é composto por dois pacotes Java: Avatar e MovimentoAvatar. O pacote Avatar contém as classes para criação do AV, incluindo criação e descrição de câmera, luz, fundo e do próprio Avatar, esquematizado na Figura 4.12, assim como as classes e os arquivos de modelagem tridimensional de cada parte que compõe o HV, conforme esquema simplificado (apenas membros do lado direito e com as partes do dedo indicador somente) apresentado na Figura 4.13.

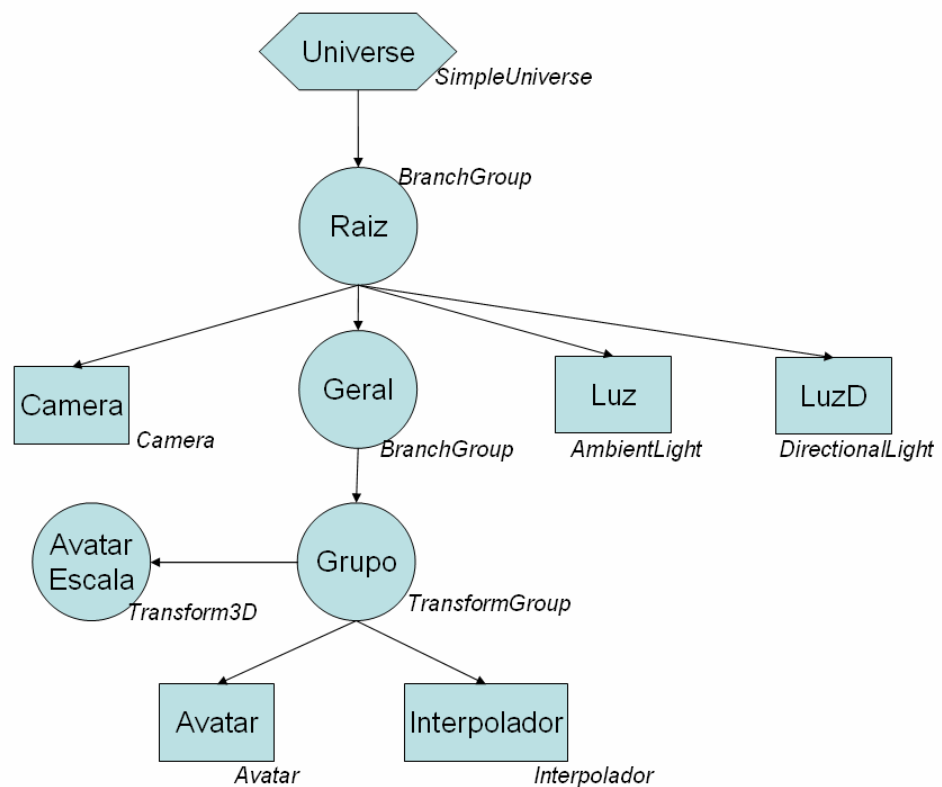


Figura 4.12 - Grafo de cena do ambiente (BAPTISTA, 2007).

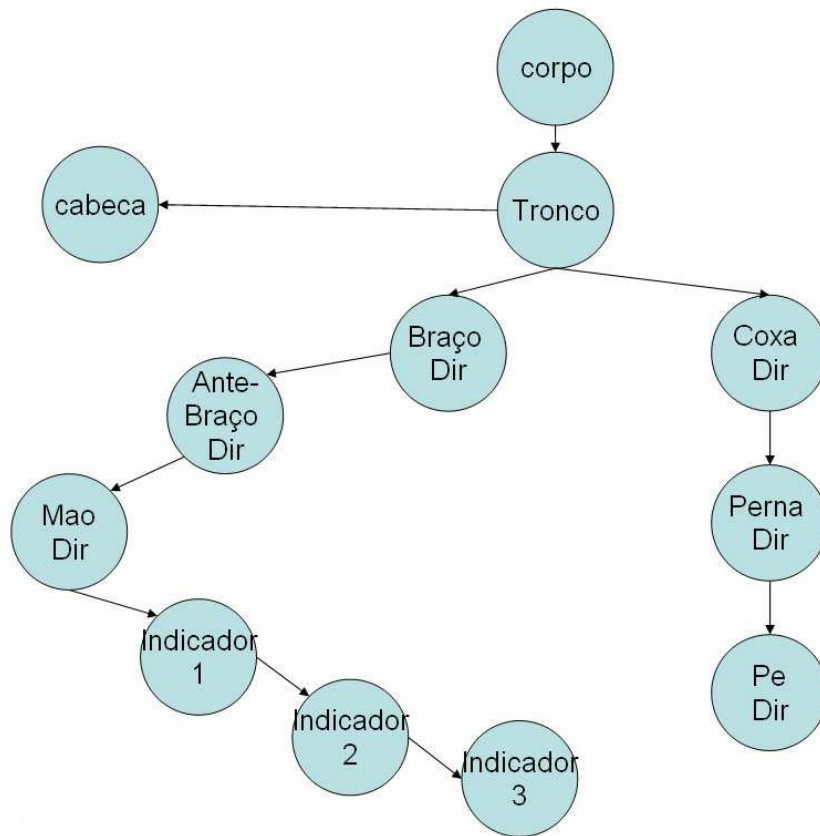


Figura 4.13: Grafo de cena do avatar (BAPTISTA, 2007).

O pacote *MovimentoAvatar* contém duas classes: uma que instancia e carrega o Avatar no sistema (*Canvas3DM*) e outra responsável pelos valores relativos aos movimentos que deverão ser executado pelo HV (*TrackerInterfaceAvatar*). Esta última possui métodos para retornar e para redefinir o ângulo atual em cada eixo para cada parte do corpo do HV. São estes os métodos chamados pelas classes de movimentação constantes nos módulos de cada dispositivo, que recebem como parâmetro o ângulo correspondente ao movimento que deve ser reproduzido no Avatar. A seção 4.2 apresenta a integração dessas classes mais detalhadamente.

4.1.5 Módulo Principal

O módulo principal é o responsável pela integração dos demais módulos no sistema. É composto de um único pacote contendo a classe principal. Uma interface gráfica foi gerada, conforme a Figura 4.14, com botões de acesso para iniciar cada módulo de

dispositivo. Ao iniciar o sistema, uma instância do AV com o Avatar é criada e exibida na tela. A cada dispositivo iniciado, essa instância é passada como parâmetro, de forma que a reprodução dos movimentos é centralizada em um único HV.

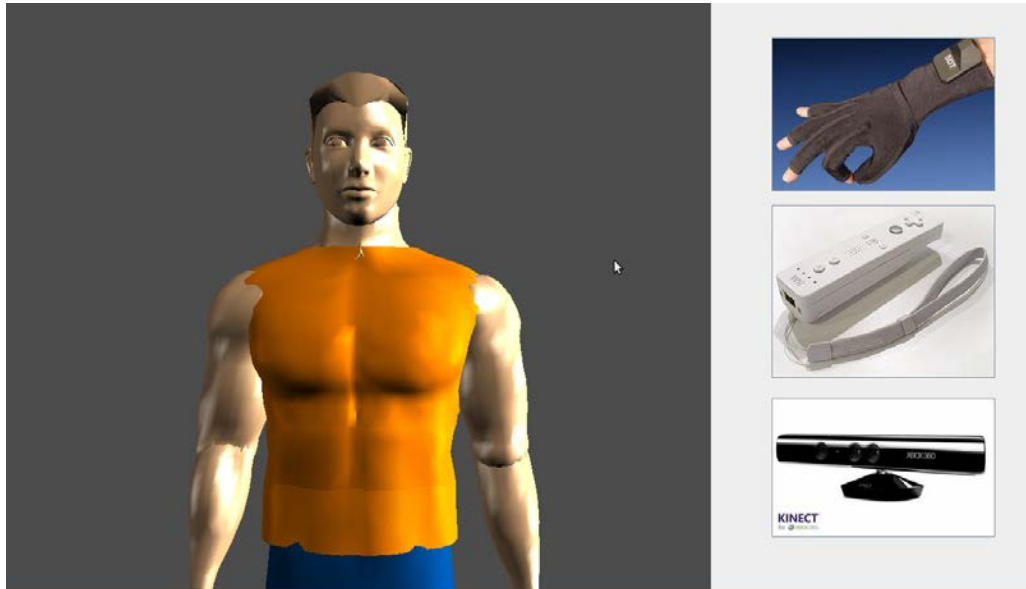


Figura 4.14 – Interface gráfica do módulo principal do sistema.

O sistema permite a inicialização dos dispositivos de forma independente, exibindo no Avatar os movimentos daqueles que estiverem ativados. O rastreamento da luva é independente dos demais, por isso pode ser reproduzido diretamente. Já o Kinect e o Wii Remote tratam de movimentos de membros em comum, e seus rastreamentos podem produzir dados conflitantes. Por isso existe a necessidade de executar uma sincronização entre os dois.

4.2 Sincronização

Cada módulo de dispositivo teve uma divisão básica em classe de interface com o usuário, classe(s) de interface com os dispositivos e classe de interface com o módulo do HV. O diagrama de atividades da Figura 5.1 apresenta um resumo do esquema de funcionamento do sistema com as etapas principais realizadas pelo usuário, sistema e dispositivos. A parte em destaque nesta figura, dentro do retângulo vermelho, é a responsável por permitir o funcionamento combinado dos dispositivos com a junção dos movimentos em um único HV

em tempo real.

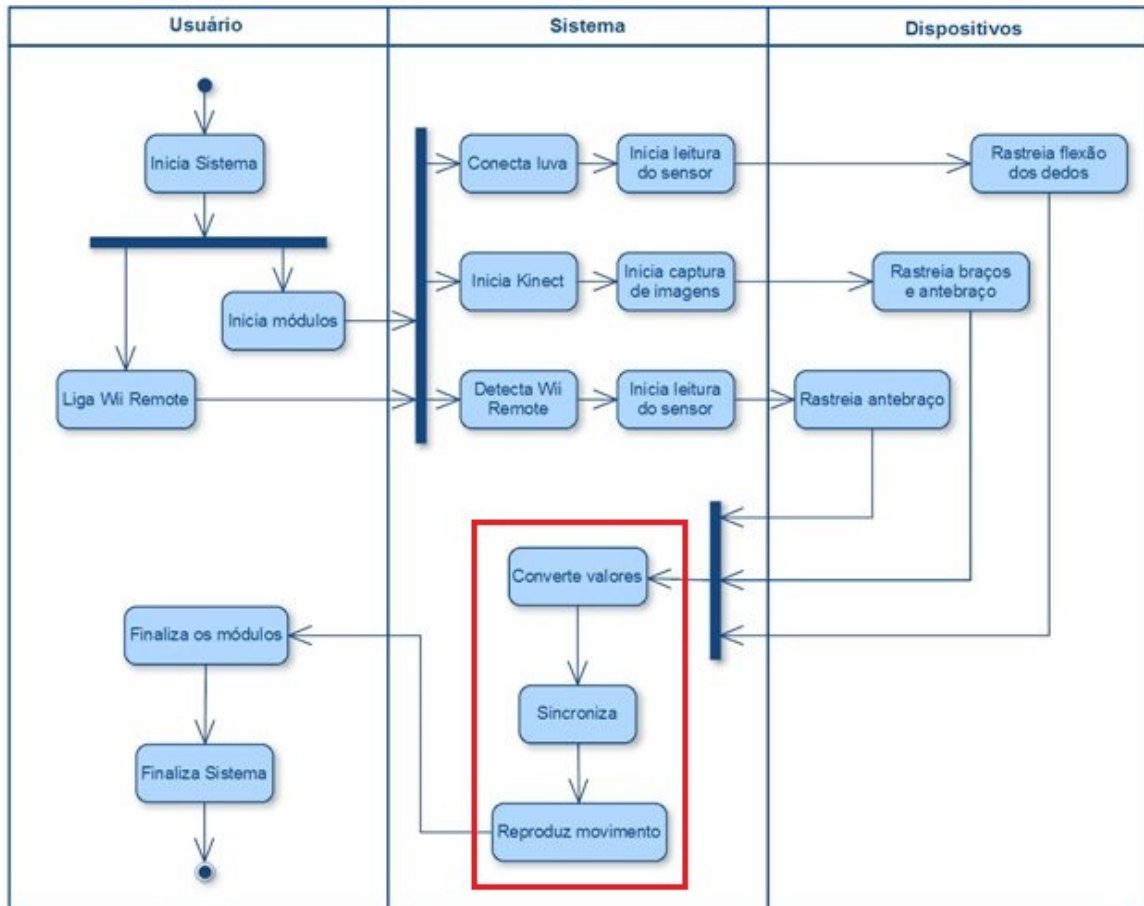


Figura 4.15 – Diagrama de atividades do sistema.

Nos pacotes dos dispositivos, as classes responsáveis pelo cálculo e conversão dos valores dos sensores, junto com a classe que envia os ângulos para o módulo do HV, permitem a movimentação correta do HV, transformando valores que podem estar em um sistema de eixos diferentes do Avatar, nos valores correspondentes ao sistema de eixos deste.

Tanto o Kinect quanto o Wii Remote podem, teoricamente, rastrear o antebraço. Porém, como já explicado no Capítulo anterior, o Kinect não obtém de forma adequada a rotação deste membro. Considerando essa informação, o rastreamento da rotação pelo sistema ficou a cargo dos Wii Remotes. Já este não capta a inclinação se o braço estiver na horizontal, pois seu valor de *pitch* não se altera. Assim sendo, a inclinação do antebraço foi obtida por meio do Kinect.

A movimentação independente da rotação do antebraço pelo Wii Remote e da inclinação do mesmo membro pelo Kinect, aliado ao fato do primeiro dispositivo não ser tão preciso e fornecer algumas vezes dados falhos por estar em posição invertida, exigiu que um

filtro para a movimentação desse membro fosse implementado. A simples junção dos dados dos dois dispositivos resultou em movimentos no Avatar que seriam impossíveis de serem realizados em um humano real, por limitações existentes nas juntas.

Foram determinadas as posições iniciais e finais que poderiam ser exibidas no Avatar e os valores equivalentes a essas posições que deveriam ser fornecidos pelos controles de cada antebraço. A partir desses dados, foram instituídos os limites inferior e superior dos dados dos controles a serem levados em consideração, descartando qualquer valor abaixo e acima deles. Valores abaixo do mínimo são exibidos como sendo na posição inicial do Avatar, acima do máximo exibe a posição final, e qualquer valor intermediário passa por um cálculo para ajustar a coordenada do controle ao sistema de coordenadas usado.

Por exemplo, enquanto o antebraço esquerdo do Avatar deve mostrar uma rotação que varia de 0° a 180° em sentido horário, o Wii Remote lê os valores de 90° a -90° para esse mesmo movimento, exigindo uma conversão no sistema de eixos, com limites para que os valores permaneçam no intervalo estipulado (Código 4.6).

Código 4.6 – Método para converter valores do Wii Remote do antebraço esquerdo para o sistema de eixos do Avatar impondo limite de movimentos.

```
...
if ( (int) Math.abs(Roll) == 180 ) {
    return (float) 90;
} else {
    if( Roll <= 180 0 && Roll > 90)
        return (float) Math.abs(Roll - 90);
    else if(Roll > 0 && Roll <= 90)
        return (float) 0;
    else if(Roll < 0 && Roll <= -90)
        return (float) Math.abs(Roll + 270);
    else
        return (float) 180;
}
...
```

Como já apresentado, uma fórmula realiza o cálculo de ângulos relativos às luvas (seção 4.1.1.1) e o Código 4.5 os cálculos relativos ao Kinect. Estes cálculos independentes consistem na primeira etapa para permitir uma saída unificada do sistema.

A etapa seguinte consiste em verificar a taxa de alteração de ângulos. Para cada eixo de cada dispositivo foi estabelecido um valor mínimo de diferença entre o valor atual e o anterior. Somente se o valor lido for maior que a diferença, a movimentação do Avatar é levada adiante e o valor anterior passa a ser o atual. Se não, o novo dado do sensor é ignorado. A interface do Wii Remote permite o controle do limite dessa diferença por meio da barra “ROLL”, conforme apresentado na Figura 4.9.

Os valores são então gravados em variáveis da classe *TrackerInterfaceAvatar*, do módulo do HV, apresentada na seção 4.1.4. Essa classe contém uma variável para cada eixo de cada parte do corpo do Avatar, e métodos *set* e *get* para cada um desses valores, além do método que calcula a matriz de rotação para cada parte baseado nos valores de cada eixo.

Antes de cada renderização do Avatar, é executado um método *pre render* que atualiza o posicionamento previsto para o Avatar, com base nessa matriz gerada anteriormente. Pelo fato dos módulos dos três dispositivos ao serem iniciados receberem como parâmetro a instância de um único Avatar, todos os valores atualizados pertencem ao mesmo HV. Dessa forma, as informações de rastreamento são unificadas de modo que os movimentos referentes aos diferentes dispositivos sejam reproduzidos de maneira sincronizada.

Na Figura 4.16 está esquematizada essa junção dos dados dos dispositivos para exibição pelo HV. Antes da renderização é executado o *pre render*, que lê os dados de *TrackerInterfaceAvatar*, atualizados pelos módulos dos dispositivos.

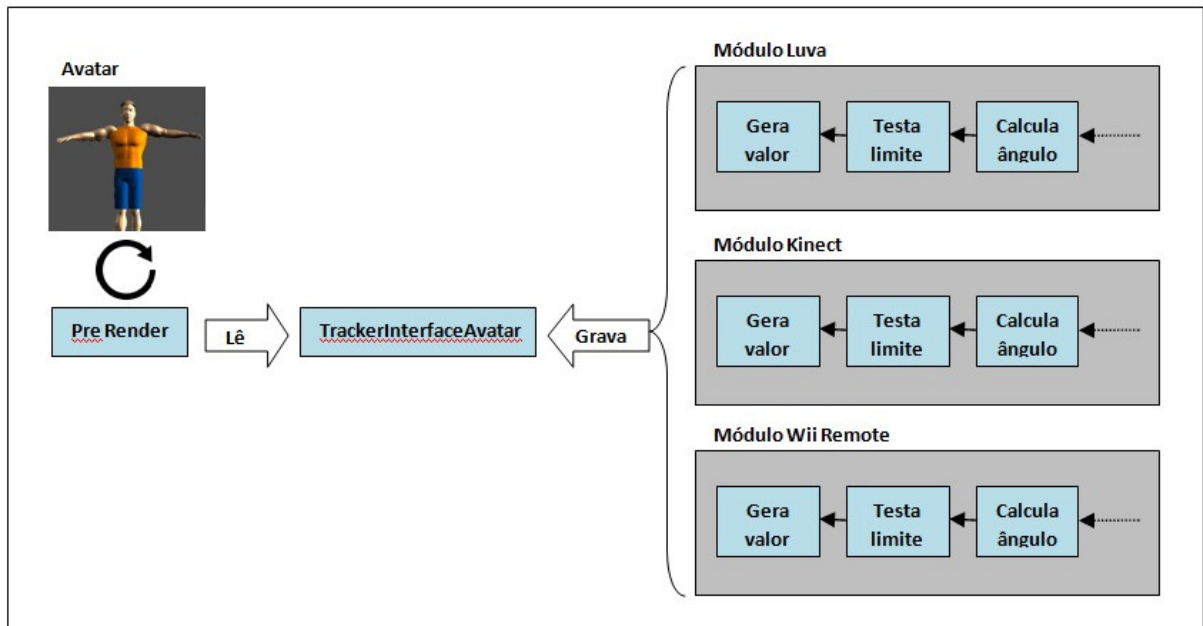


Figura 4.16 – Esquema de sincronização dos dados do sistema.

Por exemplo, para a movimentação do braço direito no eixo X (*pitch*), dentro do pacote MMSKinect a *thread* de leitura do Kinect obtém a orientação da junta do ombro, os ângulos são extraídos e os valores enviados para a instância da classe *MovimentaBracos*, conforme trecho de Código 4.7.

Código 4.7 – Obtenção, tratamento e envio de dados por KinectUserTraker.

```
...

matrix3dOmbroD      =      _SkeletonJointOrientation.get(UserID).get(
SkeletonJoint.RIGHT_SHOULDER);

ExtraiAngulos(matrix3dOmbroD);

movimentaBracoD.MoveBracoD((float)Math.toDegrees(AngX),
(float)Math.toDegrees(AngY), (float)Math.toDegrees(AngZ));

...
```

Essa classe verifica se a diferença entre o valor lido do sensor e o anterior é maior que o limite de diferença estabelecido, e em caso positivo, determina quantos graus foram movidos desde a leitura anterior e envia os novos valores para a *TrackerInterfaceAvatar* (Código 4.8), que redefine os valores a serem usados na pré-renderização (Código 4.9).

Código 4.8 – Definição do movimento a ser reproduzido no Avatar.

```

...
if ( (Math.abs(a_PICHT_Braco - PICHT_Braco) > LimPICH) ) {
    passoPICH_Braco = a_PICHT_Braco - PICHT_Braco;

    this.trackerInterfaceAvatar.setRIGHT_SHOULDER(
        this.trackerInterfaceAvatar.getX_RIGHT_SHOULDER() +
            (float) Math.toRadians(passoPICH_Braco),
        this.trackerInterfaceAvatar.getY_RIGHT_SHOULDER(),
        this.trackerInterfaceAvatar.getZ_RIGHT_SHOULDER());
}
...

```

Código 4.9 – Métodos de TrackerInterfaceAvatar usados para a pré-renderização do HV.

```

...
public void setRIGHT_SHOULDER(float x_RIGHT_SHOULDER, float
y_RIGHT_SHOULDER, float z_RIGHT_SHOULDER) {

    this.x_RIGHT_SHOULDER = x_RIGHT_SHOULDER;
    this.y_RIGHT_SHOULDER = y_RIGHT_SHOULDER;
    this.z_RIGHT_SHOULDER = z_RIGHT_SHOULDER;

}

public Matrix3f getRIGHT_SHOULDER() {

    matrixX.setZero();
    matrixY.setZero();
    matrixZ.setZero();

    matrixX.rotX(this.x_RIGHT_SHOULDER);
    matrixY.rotY(this.y_RIGHT_SHOULDER);
    matrixZ.rotZ(this.z_RIGHT_SHOULDER);

    matrixX.mul(matrixY);
    matrixX.mul(matrixZ);

    return matrixX;
}
...

```

Neste Capítulo foi apresentado o sistema desenvolvido e sua utilização. No Capítulo a seguir serão apresentados os resultados obtidos para validar a proposta de rastreamento de gestos.

5 Resultados

Neste Capítulo são apresentados os resultados obtidos com o desenvolvimento deste trabalho. Alguns testes foram efetuados para verificar o rastreamento isolado e conjunto dos dispositivos. O usuário foi filmado com uma webcam posicionada ao lado do Kinect, com resolução de 640 x 480 pixels a 30 quadros por segundo e taxa de comunicação de 60Hz.

Segundo a ferramenta de contagem de frequência de quadros presente no protótipo do HV, a movimentação do mesmo foi realizada com uma média de 29,8 quadros por segundo. A imagem do usuário captada via webcam foi posicionada ao lado da imagem do Avatar na tela do computador, e esta foi gravada utilizando-se uma ferramenta de captura de desktop a uma taxa de atualização de 30 quadros por segundo.

5.1 Resultados da luva

O uso combinado da luva permitiu capturar a flexão dos dedos, independente do posicionamento da mão no espaço, sem os problemas de oclusão que poderiam ocorrer com o uso de uma câmera para rastrear as mãos ou a necessidade de marcadores óticos.

A Figura 5.1 apresenta quadros do teste do módulo da luva em conjunto com o Avatar. Nela são reproduzidos quadros captados do rastreamento sendo realizado no usuário, e o respectivo gesto reproduzido no Avatar.

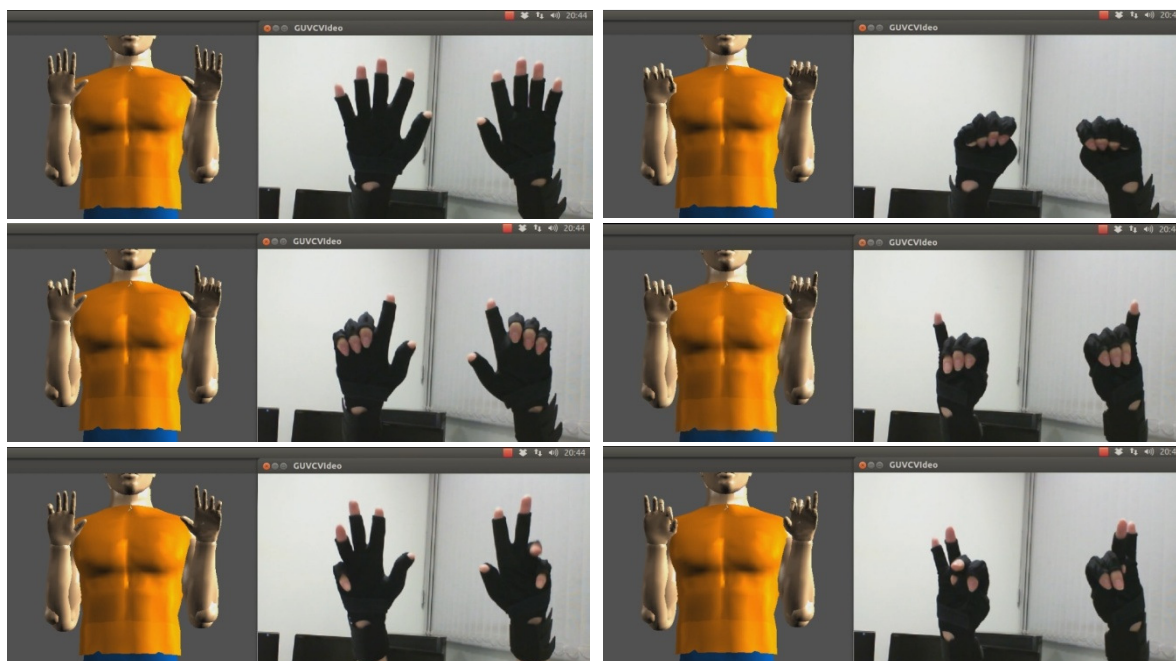


Figura 5.1 – Teste de movimentação do Avatar com a luva.

Uma falha aparente encontrada no rastreamento do sistema é com relação a uma característica da 5DT Data Glove Ultra. Alguma distorção no rastreamento da luva pôde ser percebida. Por ser confeccionada em material flexível, a flexão de um dedo pode repuxar o tecido e, consequentemente, a fibra ótica do dedo adjacente, dando a falsa leitura de que este está flexionado.

Contudo, esta falha pode ser minimizada colocando um intervalo de valores próximo a 0.0 que, ao serem lidos, devem indicar no Avatar que tal dedo está estendido.

5.2 Resultados do Wii Remote

A sensibilidade do sensor inercial do Wii Remote e seu uso invertido, com os botões virados para baixo e em dois membros com movimentos inversos, fez com que a imposição de limites aos ângulos repassados para o Avatar fosse necessária, para que o movimento reproduzido estivesse dentro dos limites de movimentação do corpo humano.

Um teste da rotação dos antebraços foi executado, realizando este movimento com os braços em posições variadas. O resultado pode ser observado nos quadros apresentados na Figura 5.2. Os quadros apresentam a rotação dos antebraços, com as palmas das mãos viradas

para o corpo na posição inicial (90° , na Figura 5.3 *a*), e em ângulos de 0° e 180° com os braços esticados para baixo (Figura 5.2 *b* e *c*), na lateral (Figura 5.2 *d* a *f*) e na frente do corpo (Figura 5.2 *g* e *h*). Como apenas o módulo do Wii Remote estava sendo usado para este teste, apenas a rotação dos antebraços foi reproduzida no Avatar, portanto os braços do mesmo aparecem em posição diferente dos braços do usuário na figura citada.



Figura 5.2 – Teste de rotação dos antebraços. Com o Wii Remote com os braços para baixo (*a*, *b* e *c*), esticados na lateral do corpo (*d*, *e* e *f*) e na frente do corpo (*g* e *h*).

A rotação dos antebraços foi realizada independente da orientação em que os mesmos ou os braços se encontravam. Porém, a sensibilidade do sensor e a falta de um filtro que uniformize os ângulos do sistema, fazem com que o movimento do Avatar oscile ocasionalmente, exibindo movimentos rápidos que não foram realizados pelo usuário.

5.3 Resultados do Kinect

Como pode ser observado nas Figuras 5.3 e 5.4, o Kinect foi capaz de rastrear os movimentos dos braços e inclinação do antebraço. A primeira Figura apresenta alguns quadros do vídeo onde foi gravado o teste com um movimento executado em 1 segundo e 15 milissegundos. A segunda Figura apresenta quadros retirados do vídeo do segundo teste, cujo movimento teve duração de 45 milissegundos.



Figura 5.3 – Primeiro teste do Kinect.



Figura 5.4 – Segundo teste do Kinect.

Ao se executar um movimento mais rápido, pode-se perceber que existe um atraso no início da movimentação do Avatar. Isto pode ser observado pela diferença entre as posições do usuário e do Avatar nos quadros da Figura 5.5. Apesar disso, a velocidade de movimentação do mesmo foi proporcional à velocidade do usuário, não aparentando haver acúmulo de atraso. A duração do movimento do Avatar no segundo teste foi de 45 milissegundos, igual a duração do movimento do usuário.

O Kinect rastreia a posição das juntas no espaço, e o Avatar trabalha com ângulos, exigindo uma conversão de um sistema para o outro. Para o trabalho desenvolvido foi usada a conversão fornecida pelo OpenNI, mas não foi realizado nenhum teste específico para verificar a precisão da orientação das juntas fornecida pelo *framework*. Essa conversão pode introduzir alguma imprecisão que resulta em atrasos na reprodução de alguns casos de pequenos deslocamentos dos membros.

5.4 Resultados da combinação

Ao realizar o rastreamento com os três dispositivos simultaneamente, não foi observado nenhum tipo de lentidão aparente significativa na movimentação do Avatar decorrente do aumento da quantidade de dados processados, além do atraso no início da movimentação já observado no teste do Kinect. O valor observado nos testes referente a este atraso entre o início do movimento do usuário e o início da reprodução no Avatar, foi de 29 milissegundos em média, de dez medições realizadas onde os valores variaram de 24 a 37 milissegundos, permitindo que a visualização fosse feita em tempo real.

A velocidade de funcionamento ao utilizar os três dispositivos simultaneamente, permitiu que nenhum tipo de sincronização prévia fosse requerida para que um determinado gesto composto de uma sequência de movimentos das mãos, braços e antebraços, fosse reproduzida no Avatar na mesma sequência.

O uso combinado do Kinect com o Wii Remote permitiu capturar a rotação do antebraço, que não é percebida com o Kinect apenas. Adicionalmente, a luva permitiu o rastreamento dos dedos, independente da posição das mãos, o que não seria possível com o Kinect por causa da ocorrência de oclusões.

A junção da calibração dos dispositivos não foi necessária. A calibração do Wii Remote se mostrou desnecessária, bastando sua ativação e conexão para que sua rotação atual fosse transmitida ao sistema. O módulo da luva foi desenvolvido com o uso da calibração continuada dos dedos, portanto sem uma única calibração inicial. O único dispositivo que exigiu tal recurso foi o Kinect.

Apesar das características citadas que podem ocasionalmente resultar em um movimento incorreto do Avatar (tecido da luva, sensibilidade do sensor do Wii Remote e orientação das juntas fornecida pelo OpenNI), o rastreamento e visualização em tempo real providos pelo sistema apresentam precisão que permite um grau de similaridade, como pode-se observar nas Figuras 5.5 à 5.9. Foram realizados quinze gestos distintos para testes, sendo alguns baseados em sinais de LIBRAS. Destes, foram selecionados 4 movimentos, cujos vídeos gravados tiveram alguns quadros selecionados e apresentados nas Figuras 5.5 à 5.8, onde poses distintas dos gestos efetuados foram comparadas. Por fim, a Figura 5.9 apresenta cinco poses estáticas.

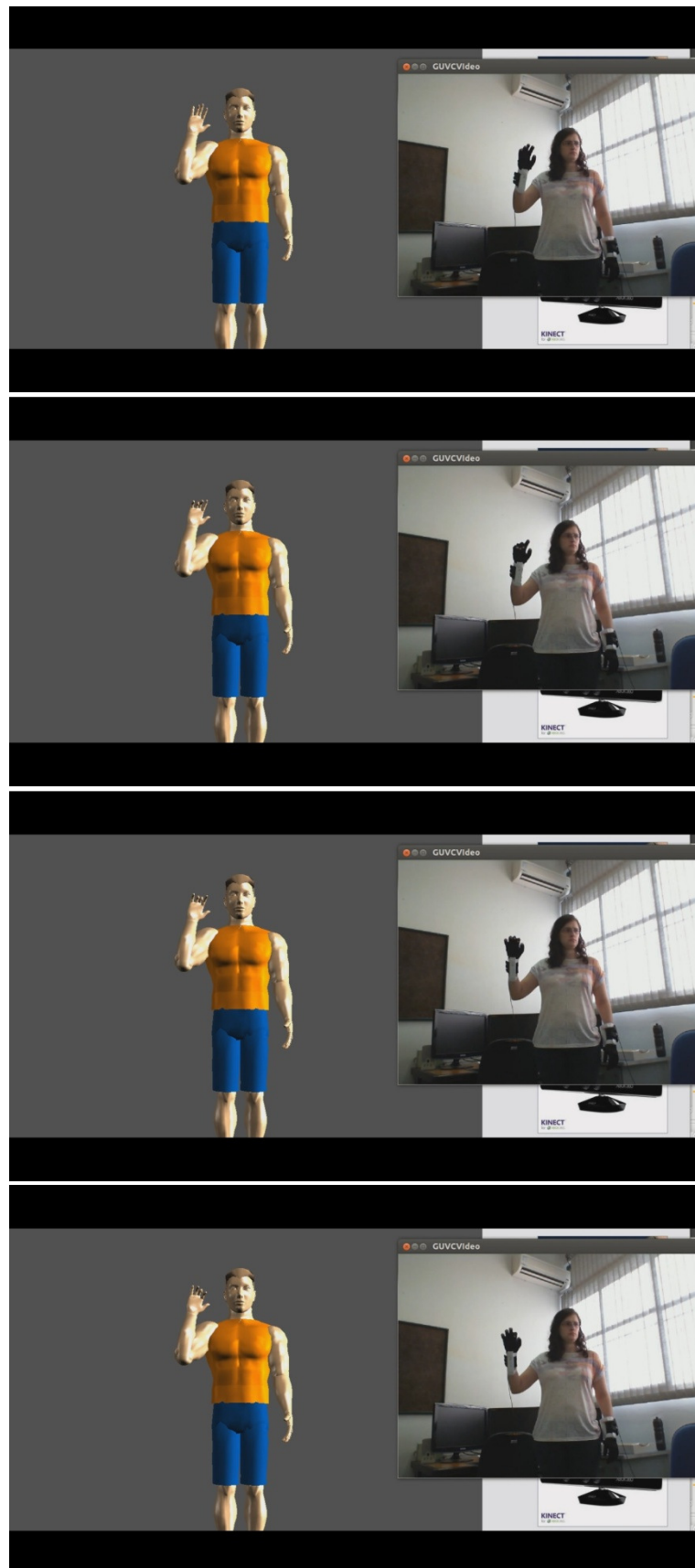


Figura 5.5 – Teste da combinação dos dispositivos dedilhando a mão direita.

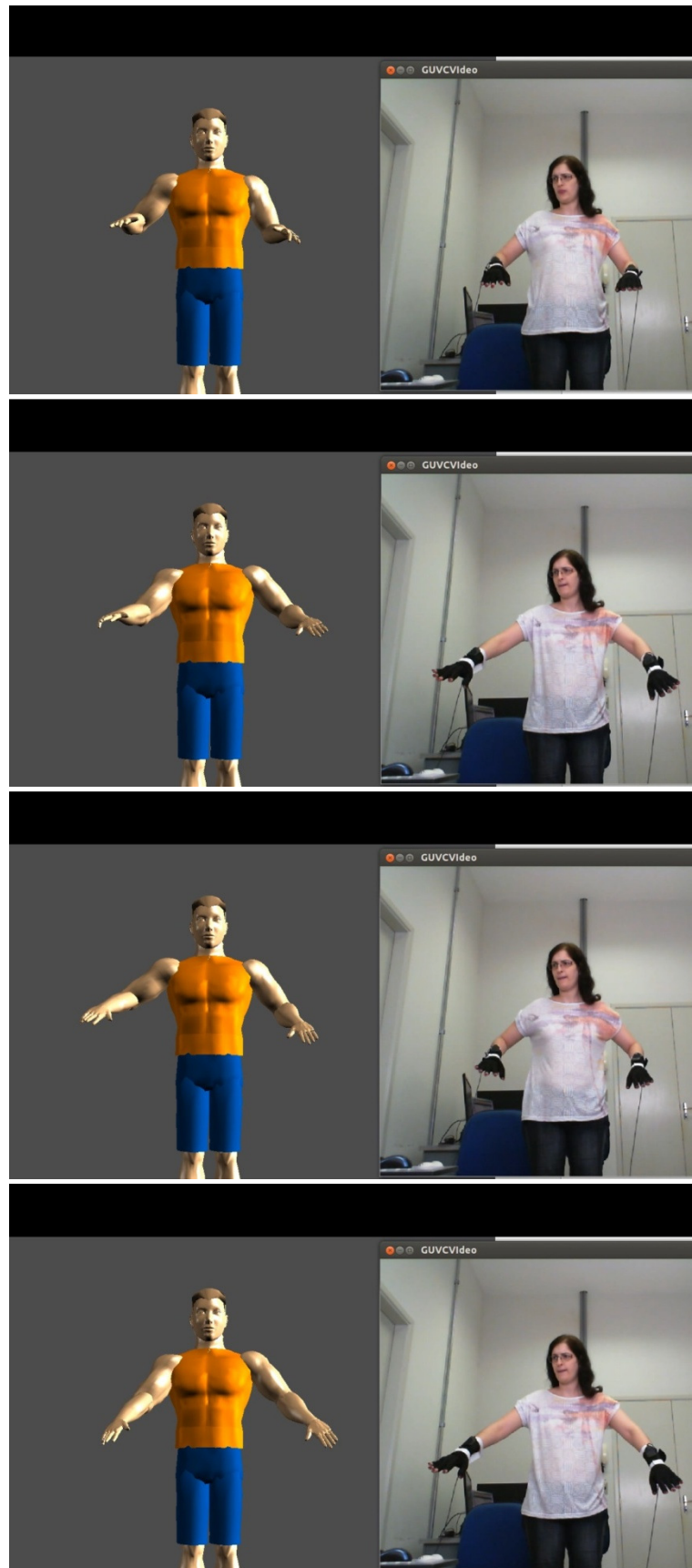


Figura 5.6 – Teste da combinação dos dispositivos com movimentos circulares das mãos em frente ao corpo, com as palmas voltadas para baixo.

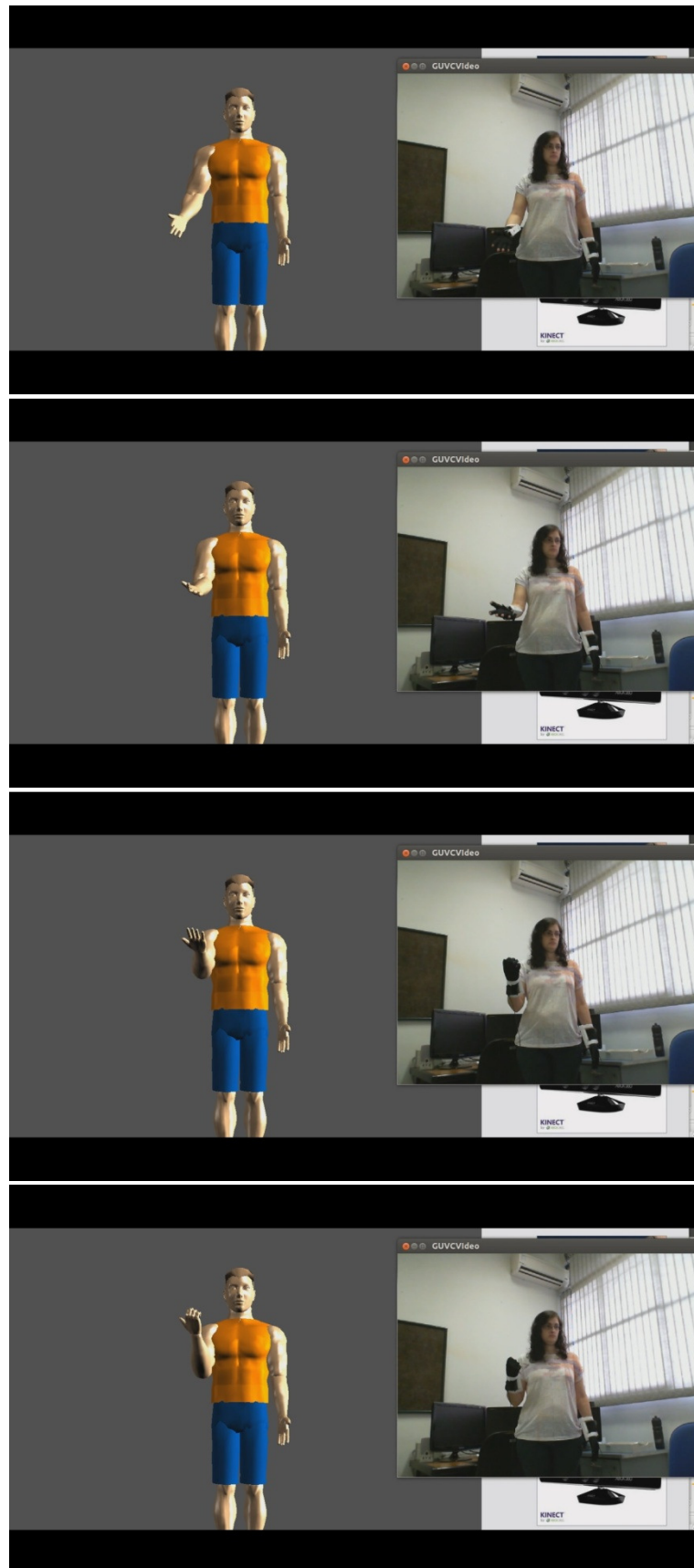


Figura 5.7 – Teste da combinação dos dispositivos fechando a mão junto ao corpo.

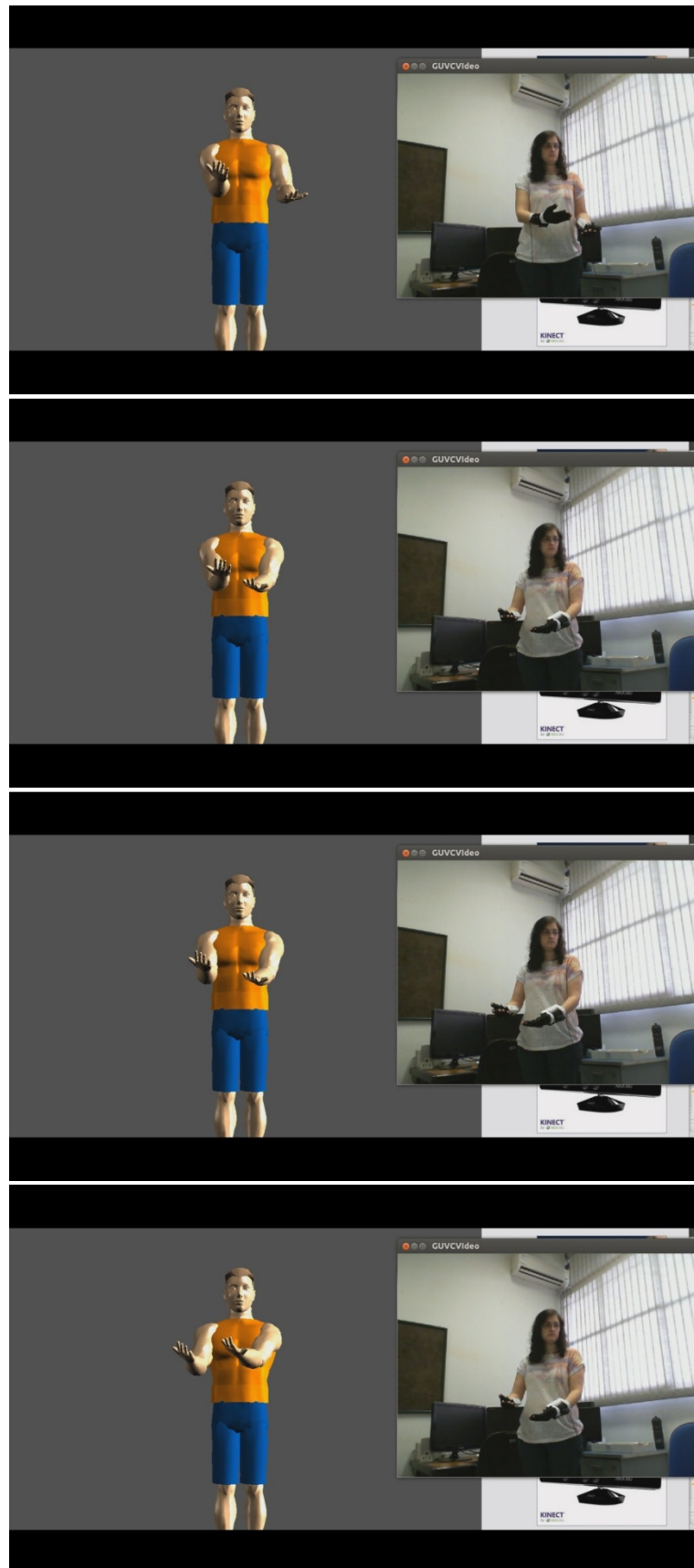


Figura 5.8 – Teste da combinação dos dispositivos movendo as mãos da esquerda para a direita com as palmas voltadas para cima.

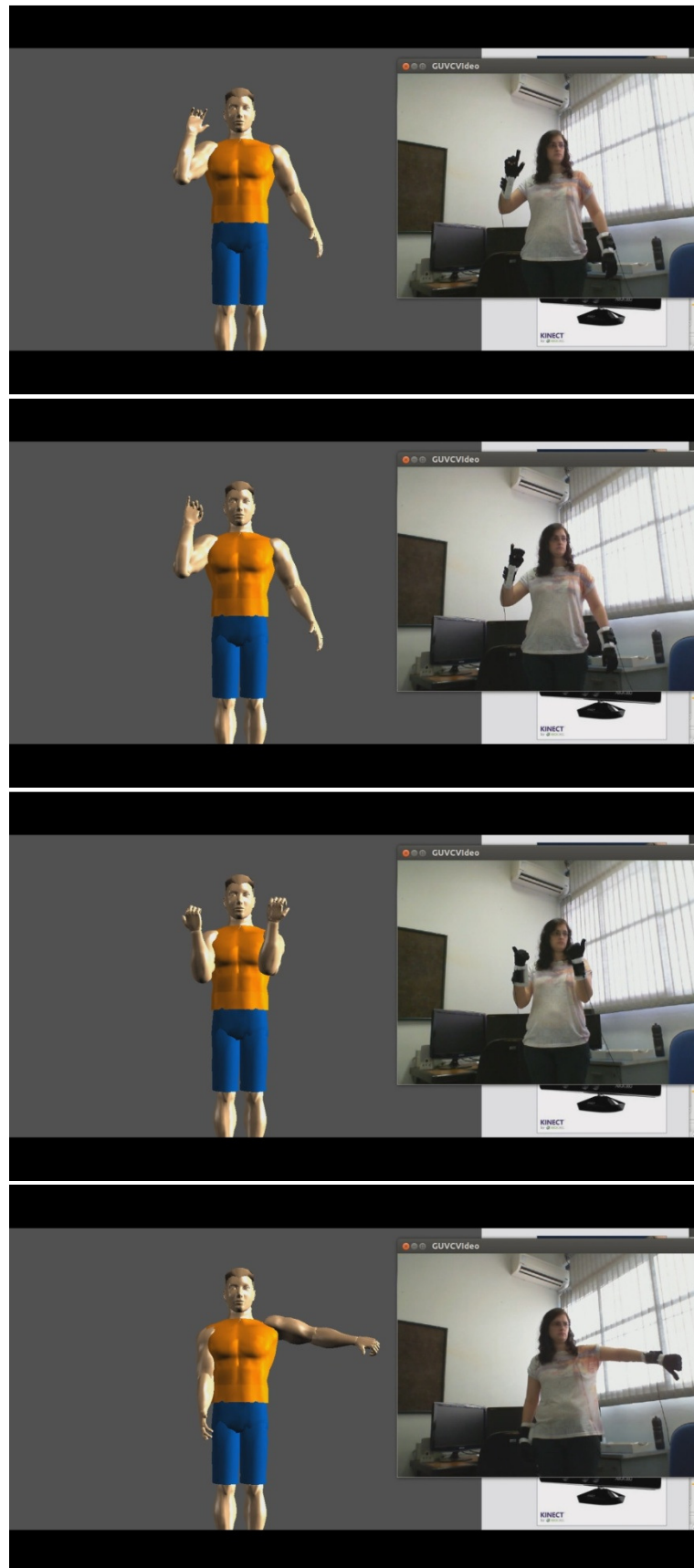


Figura 5.9 – Teste da combinação dos dispositivos para as poses estáticas indicando a letra “L” e “T” em LIBRAS, polegar estendido nas duas mãos, e apenas na mão esquerda, voltado para baixo e com o braço esticado.

Nenhuma interferência significativa visível foi observada no rastreamento do Kinect pela presença das luvas, seus cabos ou controles Wii Remote, mantendo o rastreamento como o feito quando apenas o Kinect foi utilizado.

6 Conclusão

O uso combinado de dispositivos de baixo custo se mostrou viável para rastreamento de gestos. Com a combinação foi possível complementar a informação dos gestos e assim permitir a reprodução simultânea dos movimentos em um Humano Virtual.

Os três dispositivos funcionaram concorrentemente sem interferência mútua, e nem a necessidade de um ambiente controlado, como iluminação diferenciada, ausência de ruídos ou ausência de dispositivos metálicos ou magnéticos. Também não foi necessário fazer medições exatas para posicionar o dispositivo ótico, tais como altura, grau de inclinação e distância do usuário. Foi suficiente apontar as câmeras na direção do usuário, que foi posicionado a uma distância dentro do limite de captura exigido pelo equipamento.

O sistema apresentou velocidade suficiente para proporcionar a reprodução dos movimentos em tempo real nos testes realizados. Como a rapidez na execução de gestos varia de um usuário para outro, é importante que o rastreamento seja feito sem interferência ou restrições quanto a esta característica dos movimentos.

Cada dispositivo do sistema possui uma frequência própria de envio de dados. Isto se dá tanto pela diferença nas tecnologias dos sensores utilizados, quanto pelos canais de transmissão dos dados dos dispositivos. Tal fato, contudo, não resultou em diferença de tempo do *feedback* visual entre os dispositivos. O processo de sincronização via software entre os três dispositivos devido à diferença de velocidade, portanto, não foi necessária.

A sincronização realizada consistiu em unificar os dados do rastreamento em um único Avatar. O uso de uma classe de interface entre os módulos dos dispositivos e o HV, além do uso da pré-renderização, permitiu o uso combinado dos dispositivos e a reprodução dos movimentos em tempo real.

Entre o Kinect e Wii Remote foi realizada a junção dos dados dos antebraços, com a aplicação de filtros como forma de minimizar falhas no rastreamento desses membros, já que ambos fornecem dados para o movimento dos mesmos. Para que a exibição do gesto captado pelos dois dispositivos fosse coerente com a limitação de juntas do corpo humano, uma comparação das posições do antebraço do Avatar e as indicadas pelo Wii Remote foi realizada, sendo os dados tratados antes de o movimento ser exibido.

A calibração foi simplificada, com relação ao que se esperava ser necessário, bastando o usuário realizá-la para uso do Kinect. Uma versão mais recente do OpenNI já permite o rastreamento sem calibração inicial.

A modularização do sistema permitiu o desenvolvimento e teste por etapas. O funcionamento independente de cada dispositivo viabilizou a organização do código de maneira que dispositivos possam ser substituídos ou alterados futuramente, para testes comparativos e melhorias do sistema.

A adaptação do protótipo do Avatar, com a inserção de uma classe de interface entre o sistema e a renderização, organizou o código para reutilização do HV em outros sistemas.

Trabalhos futuros

Como trabalhos futuros pretende-se testar a viabilidade de unir dados do antebraço sobre inclinação, do Wii Remote e do Kinect, com o auxílio de um detector de infravermelho para as situações em que os braços estiverem na posição horizontal.

Os ângulos absolutos lidos poderão ser tratados e refinados para aproximar o movimento a ser exibido com o movimento realizado por um humano real, incluindo o acréscimo de limitações de mobilidade das juntas do braço do HV, de acordo com os limites da anatomia humana.

Um comparativo poderá ser realizado entre o sistema desenvolvido e o de Marca (2011), que faz uso apenas de Wii Remotes. Um sistema puramente ótico também poderá ser comparado.

Será verificada a viabilidade de uso de dois Kinects, um para rastreamento dos braços e o outro para o rastreamento das mãos. O segundo dispositivo seria responsável pelo rastreamento atualmente realizado pelas luvas.

A gravação de movimentos em um Banco de Dados será habilitada para permitir o uso da ferramenta em outros sistemas, como um dicionário de sinais ou, com auxílio de uma

ferramenta de inteligência artificial para reconhecimento de padrões, em reconhecimento de gestos, possibilitando o uso da ferramenta no ensino e treinamento de atividades e tradução de gestos.

Referências

AGOSTINHO, I. A.; BREGA, J. R. F.; DIAS, D. R. C. **Combinação de dispositivos de baixo custo para rastreamento de gestos.** In: VIII Workshop de Realidade Virtual e Aumentada (WRVA 2011), Uberaba, 2011.

AILIVE. *LiveMMove 2 Director's Cut.* Disponível em: <www.ikuni.com/papers/directorsManual2_en.pdf>. Acesso em 07 de junho de 2012.

ANTONIAC, P. M. **New user interfaces for mobile devices using augmented reality :expanding the interaction by intuitive gesture recognition.** VDM Verlag Dr. Müller. 2008.

ASCENSION TECHNOLOGY CORPORATION. **MotionStar Real-Time Motion Capture.** Product brochure. Disponível em: <http://www.ascension-tech.com/docs/products/motionstar_10_04.pdf>. Acesso em 14 setembro 2013.

AVIZZANO, C. A. et al. **A navigation interface based on head tracking by accelerometers.** In: 13th IEEE International Workshop on Robot and Human Interactive Communication, 2004.

BAO, P. T.; BINH, N. T.; KHOA, T. D. **New Approach to Hand Tracking and Gesture Recognition by a new Feature Type and HMM.** In: Proceedings of the 6th international conference on Fuzzy systems and knowledge discovery - Volume 4, Piscataway. 2009.

BAPTISTA, F. **F-LIBRAS - Ambiente Integrado de Ensino-Aprendizagem para Língua Brasileira de Sinais.** Dissertação de Mestrado, Univem. Marília. 2007.

BHUIYAN, M.; PICKING, R. **Gesture-controlled user interfaces, what have we done and what's next?** In: Proceedings of the Fifth Collaborative Research Symposium on Security, E-Learning, Internet and Networking (SEIN 2009), Darmstadt, Alemanha, 26-27 Novembro 2009, pp59-60.

BLUTEAU, J. et al. **Haptic guidance improves the visuo-manual tracking of trajectories.** In: PLoS ONE 3, 3 (2008) e1775 (Journal). 2008.

BREGA, J. R. F. **Análise e Implementação de um Sistema de Realidade Virtual de Ensino-Aprendizagem para Língua Brasileira de Sinais.** Tese de livre docência. Bauru, 2010.

BREGA, J. R. F.; KELNER, J. (Org.) **Interação em Realidade Virtual e Realidade Aumentada**. Canal 6, Bauru. 2010.

BURDEA, G. C.; COIFFET, P. **Virtual Reality Technology**. 2a ed. Wiley-Interscience. 2003.

COLANTONIO, S. et al. **Object tracking in a stereo and infrared vision system**. *Infrared Physics & Technology*. Volume 49, p. 266-271, 2006.

DAVIS, J.; SHAH, M. **Gesture Recognition**. Relatório Técnico. University of Central Florida. 1994.

DONG, L.; WU, J.; CHEN, X. **A body activity tracking system using wearable accelerometers**. In: IEEE International Conference on Multimedia and Expo, p. 1011-1014, 2007.

DORFMUELLER-UHAAS, K; SCHMALSTIEG, D. **Finger tracking for interaction in augmented environments**. In *Proc. IEEE International Symposium on Augmented Reality*. IEEE Press, 2001.

DORFMÜLLER-UHAAS, K.; WIRTH, H. **Real-Time Hand and Head Tracking for Virtual Environments Using Infrared Beacons**. In: Proceedings of the International Workshop on Modelling and Motion Capture Techniques for Virtual Environments, Londres. 1998.

DUCHÉ, G. **WiiuseJ**. Disponível em: <<https://code.google.com/p/wiiusej/>>. Acesso em: 21 jul 2013.

EBERLY, D. **Euler Angle Formulas**. Geometric Tools, LLC. 2011.

ERİŞMİŞ, M. A. **MEMS Accelerometers and Gyroscopes for Inertial Measurement Units**. Dissertação de mestrado. Middle East Technical University, Turkia. 2004.

FENG, Z. et al. **A Novel Tracking Method Based on Optimized Selection**. In: 2nd International Congress on Image and Signal Processing, 2009.

FIFTH DIMENSION TECHNOLOGIES – 5DT. **5DT Glove Ultra Series User's Manual**. Disponível em: <[http://www.5dt.com/downloads/dataglove/ultra/5DT Data Glove Ultra Manual v1.3.pdf](http://www.5dt.com/downloads/dataglove/ultra/5DT_Data_Glove_Ultra_Manual_v1.3.pdf)>. Acesso em 06 jun 2011.

FOURSA, M. **Real-time infrared tracking system for Virtual Environments.** In: Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry, Nova Iorque. 2004.

FUA, P. et al. Body Cloning and Body Motion Capture. In: THALMANN, N. M.; THALMANN, D. (Ed.) **Handbook of Virtual Humans.** John Wiley & Sons. Inglaterra. 2004 p. 52-74.

GHIROTTI, S. E.; MORIMOTO, C. H. **Um sistema de interação baseado em gestos manuais tridimensionais para ambientes virtuais.** In: Anais do IX Simpósio de Fatores Humanos em Sistemas Computacionais - IHC 2010. Belo Horizonte, 2010.

H-ANIM HUMANOID ANIMATION WORKING GROUP. **Specification for a standard Humanoid.** 2002. Disponível em: <<http://www.h-anim.org/Specifications/H-Anim1.1>>. Acesso em 26 fev 2014.

HARUNA, N. C. A. **Teorema de Thales: Uma abordagem do processo ensino-aprendizagem.** Dissertação de Mestrado em Educação Matemática. Pontifícia Universidade Católica de São Paulo – PUC –SP. 2000.

HOLDEN, E.; OWENS, R.; ROY, G. G. **3D Hand Tracker for Visual Sign Recognition.** Relatório Técnico. 99/2, Department of Computer Science, The University of Western Australia. 1999.

HUYGHE, B.; DOUTRELOIGNE, J.; VANFLETEREN, J. **3D orientation tracking based on unscented Kalman filtering of accelerometer and magnetometer data.** In: IEEE Sensors Applications Symposium, 2009.

JHA, A. R. **Infrared Technology.** John Wiley & Sons, 2000.

KEIR, M.S. et al. **A New Approach to Accelerometer-based Head Tracking for Augmented Reality & Other Applications.** In: IEEE International Conference on Automation Science and Engineering, 2007.

KINECT. **Kinect.** Disponível em: <<http://www.xbox.com/en-US/kinect>>. Acesso em 06 jun 2011.

KING, R. C. et al. **An HMM Framework for Optimal Sensor Selection with Applications to BSN Sensor Glove Design.** In: Proceedings of the 4th workshop on Embedded networked sensors, Nova Iorque. 2007.

KOSCHAN, A.; ABIDI, M. **Digital Color Image Processing**. John Wiley & Sons, 2008.

KÜHN, T. **The Kinect Sensor Platform**. In: Proc. Advances in Media Technology. Technische Universität München, 2011, pp. 1–4.

LEE, J. C. **Hacking the Nintendo Wii Remote**. In: Pervasive Computing, IEEE, 2008.

LIBRAS.ORG.BR. **O que é LIBRAS – Informações técnicas**. Disponível em: <<http://www.libras.org.br/libras.php>>. Acesso em: 25 jul 2011.

MAEDA, M. et al. **Tracking of user position and orientation by stereo measurement of infrared markers and orientation sensing**. Proceedings of the Eight Int. Symposium on Wearable Computers. Vol.1. 2004.

MARCA, A. F. L. **Sistema Gerador de Apoio de um Dicionário Temático Visual-Gestual Baseado em Realidade Virtual**. Dissertação de Mestrado. Bauru, 2011.

MICROSOFT. **Kinect for Windows Sensor**. Disponível em: <<http://msdn.microsoft.com/en-us/library/hh855355.aspx>>. Acesso em 02 fev 2013.

MITRA, S.; ACHARYA, T. **Gesture Recognition: A Survey**. In: IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 37, 2007, pp. 311-324.

MOESLUND, T. B.; GRANUM, E. **A Survey of Computer Vision-Based Human Motion Capture**. In: Computer Vision and Image Understanding 81. 2001.

NETTO, A. V.; MACHADO, L. S.; OLIVEIRA, M. C. F. **Realidade Virtual**. Visual Books, 2002.

NGUYEN, K. D. et al. **A Wearable Sensing System for Tracking and Monitoring of Functional Arm Movement**. In: IEEE/ASME Transactions on Mechatronics. 2011.

NINTENDO. **Controllers at Nintendo**. Disponível em: <<http://www.nintendo.com/wii/console/controllers>>. Acesso em 06 jun 2011.

OGRIS, G. et al. **Using Ultrasonic Hand Tracking to Augment Motion Analysis Based Recognition of Manipulative Gestures**. In: Proceedings of the Ninth IEEE International Symposium on Wearable Computers, Washington. 2005.

PAN, Z. et al. **A Real-time Multi-cue Hand Tracking Algorithm Based on Computer Vision.** In: In Proceedings of Virtual Reality Conference. 2010.

PARK, I.; KIM, J.; HONG, K. **An Implementation of an FPGA-Based Embedded Gesture Recognizer Using a Data Glove.** In: Proceedings of the 2nd international conference on Ubiquitous information management and communication, 2008.

PIMENTEL, K; TEIXEIRA, K. **Virtual Reality – through the new looking glass.** 2a ed.. New York, McGraw-Hil, 1995.

PINHANEZ, C. Interfaces não-convencionais. In: TORI, R.; KIRNER, C.; SISCOUTO, R. (Ed.) **Fundamentos e tecnologia de realidade virtual e aumentada.** Editora SBC. Porto Alegre, 2006.

PITERI, M. A.; RODRIGUES, J. C. (Coord.) **Fundamentos de Visão Computacional.** Viena. Presidente Prudente, 2011.

POLHEMUS. **Fastrak.** Disponível em: <www.polhemus.com>. Acesso em 27 de set de 2013.

PRIMESENSE. **OpenNI.** Disponível em: <<http://75.98.78.94/>> . Acesso em: 23 fev 2012.

PUSTKA, D; KLINKER, G. **Dynamic gyroscope fusion in Ubiquitous Tracking environments.** In: Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality. 2008.

QUEK, F. K. **Eyes in the interface.** In: Image and Vision Computing , 13, 1995, pp. 511 - 525.

SATO, K. **Haptic Telexistence.** In: 34th International Conference and Exhibition on Computer Graphics and Interactive Techniques, SIGGRAPH2007. San Diego, 2007.

SATOH, K.; UCHIYAMA, S.; YAMAMOTO, H. **A Head Tracking Method Using Bird's-Eye View Camera and Gyroscope.** In: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality, 2004.

SCHLÖMER, T. et al. **Gesture Recognition with a Wii Controller.** In: Proceedings of the 2nd International Conference on Tangible and Embedded Interaction , 2008, pp. 11-14.

SEGEN, J.; KUMAR, S. **Human-Computer Interaction using Gesture Recognition and 3D Hand Tracking**. In: International Conference on Image Processing, 1998. ICIP 98. Proceedings. 1998.

SHERMAN, W.R.; CRAIG, A. B. **Understanding Virtual Reality**. Morgan Kaufman Publishers, 2003.

STIEFMEIER, K. et al. **Combining Motion Sensors and Ultrasonic Hands Tracking for Continuous Activity Recognition in a Maintenance Scenario**. In: 10th IEEE International Symposium on Wearable Computers (ISWC). Montreux, Suíça, 2006.

STURMAN, D. J.; ZELTZER, D. **A survey of glove based input**. In: IEEE Computer Graphics and Applications. Volume 14 Issue 1, 1994.

SUN, **Java Native Interface**. Disponível em: <java.sun.com/developer/onlineTraining/Programming/JDCBook/jni.html>. Acesso em: 12 jul 2012.

SUN, **Java 3D**. Disponível em: <<http://www.oracle.com/technetwork/java/javase/tech/index-jsp-138252.html>> . Acesso em 8 jun 2013.

TEICHRIEB, V.; FIGUEIREDO, L. S. Interação natural. In: BREGA, J. R. F.; KELNER, J. (Org.) **Interação em Realidade Virtual e Realidade Aumentada**. Canal 6, Bauru. 2010 p. 9-15.

TORI, R.; KIRNER, C.; SISCOUTO, R. (Ed.) **Fundamentos e tecnologia de realidade virtual e aumentada**. Editora SBC. Porto Alegre, 2006.

TRAINA, A. J. M.; OLIVEIRA, M. C. F. **Apostila de Computação Gráfica**. USP, 2006.

VRDEPOT **CyberGlove Specification**. Disponível em: <<http://www.vrdepot.com/trkglv.htm>>. Acesso em 02 ago 2011.

WACHTER, S.; NAGEL, H. **Tracking Persons in Monocular Image Sequences**. In: Computer Vision and Image Understanding, Journal, Volume 74 Issue 3, June 1, Nova Iorque. 1999.

WINGRAVE, C. A. et al. **The Wiimote and Beyond: Spatially Convenient Devices for 3D User Interfaces**. In: IEEE Computer Graphics and Applications, 2010.

WU, C.; CHANG, Y.; TSENG, Y. **On Optimization of Accelerometers Deployment for Human Posture Tracking.** In: International Conference on Body Sensor Networks (BSN), 2011.

XBOX. **Manual e Garantia do Sensor Kinect.** Disponível em: <<http://download.microsoft.com/download/F/6/6/F6636BEB-A352-48EE-86A3-ABD9C0D4492A/kinectmanual.pdf>>. Acesso em 02 fev 2013.

XSENS. **MTi OEM – Product Specifications.** Disponível em: <<http://www.xsens.com/en/general/mti-oem>>. Acesso em 03 ago 2011.

YEOH, W. et al. **Real-time tracking of flexion angle by using wearable accelerometer sensors.** In: 5th International Summer School and Symposium on Medical Devices and Biosensors, 2008.

YOON, H. et al. **Human Computer Interface for Gesture-Based Editing System.** In: ICIAP '99 Proceedings of the 10th International Conference on Image Analysis and Processing. 1999.

ZHANG, Z. **Microsoft Kinect Sensor and Its Effect.** In: IEEE Multimedia, 19, 2012, pp. 4-10.

ZHOU, H.; HU, H. **A Survey - Human Movement Tracking and Stroke Rehabilitation.** Universidade de Essex, Inglaterra. ISSN 1744-8050, 2004.

ANEXO A – Tecnologias e pesquisas com GCUI

Tabela A.1 - Projetos comerciais com tecnologia voltada à GCUI (BHUIYAN; PICKING, 2009).

Name	Applications	Technology	Interface	Functionalities	Uses	Products
EYE TOY (Sony) -2003	Interacting as personalized gamer with Sony play station games.	Color USB digital camera device using gesture recognition and sound of microphone.	Game on TV, PC etc.	Uses computer vision & Gesture recognition to process images taken by the camera	Gaming application	Camera with console
Wii Nintendo[8] -2006	Wireless and motion-sensitive remote with game console	Game with any TV, computer etc.	Interface in the screen	Remote offers an intuitive, natural way to play games.	Gaming application control	Game console and remote control
XBOX live vision -2006	Interacting as personalized gamer with Microsoft's XBOX-360 games.	Color USB digital camera device using gesture recognition and sound of microphone.	Game on tv, pc etc.	Uses computer vision & Gesture recognition to process images taken by the camera	Gaming application	Camera with console
http://www.gesturetek.com/[25] - 2008	Controlling pc, mobile or console application using camera or phone.	3D Camera for computer vision. camera in mobile device. pointing over frame	Mobile , pc, large screen, table top interface.	Capturing gestures for normal PC operation. Body gesture based gaming in the mobile device	More intuitive vision activities without keyboard & mouse	Natural input for different media, PC, mobile, screen etc.
/www.mgestyk.com -2009	Interaction with computer to operate games and application	Software for hand-gesture processing & 3D camera	PC based interface	Capture hand movements and translate them into commands for controlling Windows application	Control games and other windows applications	Camera and software

Tabela A.2 - Pesquisas envolvendo tecnologia GCUI (BHUIYAN; PICKING, 2009).

Research Study-year	Applications	Users-Gesture	Technology	Interface	Issues addressed	Result /Conclusion
"Put-That-There": Voice and Gesture[15]-1980	Pointing to items on the large screen with voice.	General-Hand	Large screen, a space-sensing cube on wrist, & microphone	Large screen display	How voice and gesture can be made to inter-orchestrate, actions	Conjoint use of voice- & gesture-recognition to command events on a large raster-scan display.
Hand drawn gesture for editing task [1]-1986	Text-editing	General-Hand	Stylus	Write on the surface of a display with a stylus	Consistency in using & gesture variability. Most common gestures Easy to use and remember	People behave in a way which makes gesture-driven interfaces feasible.
A hand gesture interface device[13]-1987	Manipulating pc objects using hand glove	General-Hand	Glove & the device incorporate a collection of technologies.	Computer screen with a 3D hand model, like mouse cursor	Manipulating 3D virtual objects with 2D controllers such as touch pads and mice are awkward.	Input devices, the Z-Glove and the DataGlove,
Charade[24]-1993	Presentation application using hand gloves	General-Hand	DataGlove with serial port of the pc.	Cursor movement in the computer screen using hand gesture.	Gestural command is represented by three icons. - distinguishes between gestural commands & other free-hand gestures.	Capture of gestures by using a DataGlove,
Camera based web interface by IBM[20]-1999	Web browsing with touch FREE switch	Disabled People-Body	Camera as input of web browser	Web interface	Real time behaviour of image processing for video. Mapping the gesture as command	Development of the TouchFREE applications.
Gesture Pendant [22] -2000	Controlling home appliances using wearable pendant	Disabled People-Hand	A small camera, part of a necklace, is ringed by IR LEDs & has a IR pass filter over lens, computer, controller device (slink-E, x10)	Control using wireless video	Used in a variety of lighting conditions. Enabling technology for elderly	Gesture controlled input device pendant can recognize 95% control gestures & 97% user defined gestures
GestureWrist and GesturePad.[19]-2001	Control any device using sensor and wrist device in the wrist or Pad.	General-Hand	Input of wearable computer	In wrist and cloths. To control any device	Input devices should be natural and unnoticeable to use in various social settings.	2 Input devices for wearable computer.
Ubi-Finger[21] - 2001	Control of appliance, presentation, window scrolling of Mobile PDA	General-Finger	Fingertip device with infrared sensor works via network.	Mobile PDA interface	Control various appliances by using existing metaphors and corporeality	Prototype systems, and evaluate how effective
XWand: UI for Intelligent Spaces[23] -2003	Interacting with intelligent environment using wireless sensor package	General-Hand	Handheld device , wand with variety of sensors & use Bayes networks	Natural interface with audio and LED feedback	Unifying the results of pointing detection and speech recognition,	Interface for intelligent environments

Tabela A.2 (continuação) - Pesquisas envolvendo tecnologia GCUI (BHUIYAN; PICKING, 2009).

Research Study- year	Applications	Users- Gesture	Technology	Interface	Issues addressed	Result /Conclusion
Visual Touchpad [6]-2004	Interaction with PCs using touchpad.	General- Hand	Quadrangle panel with a rigid backing with PCs & two cameras.	on PCs, large wall displays	Visualization of gesture in the screen. Single and 2 handed gesture commands.	Vision-based input device that allows for fluid two-handed gestural interactions
Accelerometer based gesture control [4]-2004	Recognizing, training customised accelerometer based gestures	Elderly- Hand	Accelerometer, ubiquitous device	Accelerometer for recognition, training	Importance of the training in gesture control interface.	To reduce the error customization and training of gesture play important role
Gesture control for a design environment [3]-2005	Not specified.	Any-any	Study only	Study only	Usefulness of the gesture modality compared to other. Difference of gestures for the same task.	Gestures are a natural modality for certain tasks & can be augmented
Visualization method [2]-2006	Gesture visualization method, animation of hand movement performed during the gesture control	Elderly- Hand	Visualization method architecture using the accelerometer data	Gesture projection of the 3D path onto a plane, video, VRML	Concepts of the gesture visualization and it could be utilized in providing essential feedback and guidance.	Visualization provides information about the gesture performed.
http://atlasgloves.org -2006	Interaction with PCs using DIY hand gloves	General	Pair of illuminating gloves to track hand gestures. The Open Source Atlas Gloves application, webcam	Application in the PC	Controlling 3D mapping applications like Google Earth .	Open source API and DIY gloves
Intelligent Smart Home Control Using Body Gestures[11] -2006	control of smart home environments such as lights and curtains	General	Smart home with 3 CCD cameras. Marker attached in the human body	Gesture extracted as 3D and 2D view.	Continuously changing gesture can be used.	Recognition rate is 95.42% for continuously changing gestures
Head gesture recognition [7] -2007	Hands free control system of an intelligent wheelchair	Elderly & disabled	Wheelchair with laptop & webcam.	Laptop Interface with wheelchair user.	Gestures are used to generate motion control commands	Head gesture control system.
Select-and-Point[17] -2008	Enables controls to applications such as MS Office, web browser & multi-media programs in multiple devices.	General	Composed of three parts-a presence server, controlling peer & controlled peer using camera, software tools	Table top, large screen PC or mobile	Eliminating different cumbersome processes in the group meeting & provides a intuitive interaction style through a pointing gesture	Implementing intelligent meeting room