

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

CAIO CARVALHO MOREIRA

**NAVEGAÇÃO, RECONHECIMENTO DE GESTOS E CONTROLE DE INTERFACE
NO SISTEMA ITV UTILIZANDO O DISPOSITIVO KINECT**

DM: 12/2013

BELÉM - PA
2013

CAIO CARVALHO MOREIRA

**NAVEGAÇÃO, RECONHECIMENTO DE GESTOS E CONTROLE DE INTERFACE
NO SISTEMA ITV UTILIZANDO O DISPOSITIVO KINECT**

Dissertação de Mestrado submetida à banca examinadora aprovada pelo colegiado do Programa de Pós-graduação em Engenharia Elétrica, sendo julgada adequada para a obtenção do grau de Mestre em Engenharia Elétrica na área de Computação Aplicada.

Orientador: Prof. Dr. Manoel Ribeiro Filho.

DM: 12/2013

BELÉM - PA.

2013

UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**“NAVEGAÇÃO, RECONHECIMENTO DE GESTOS E CONTROLE DE INTERFACE
NO SISTEMA ITV UTILIZANDO O DISPOSITIVO KINECT”**

AUTOR: CAIO CARVALHO MOREIRA

DISSERTAÇÃO DE MESTRADO SUBMETIDA À BANCA EXAMINADORA APROVADA PELO COLEGIADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA, SENDO JULGADA ADEQUADA PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA NA ÁREA DE COMPUTAÇÃO APLICADA.

APROVADA EM: **10 / 05 / 2013**

BANCA EXAMINADORA:

Prof. Dr. Manoel Ribeiro Filho

(Orientador - PPGEE/UFPA)

Prof. Dr. Eloi Luis Favero

(Membro - PPGEE/UFPA)

Prof. Dr. Bianchi Serique Meiguins

(Membro - PPGCC/UFPA)

VISTO:

Prof. Dr. Evaldo Gonçalves Pelaes

(Coordenador do PPGEE/ITEC/UFPA)

RESUMO

Esta dissertação apresenta a implementação de navegação no ambiente virtual, reconhecimento de gestos e controle de interface, feitos através do dispositivo Kinect, no Sistema ITV: um sistema de treinamento de operadores e mantenedores de usinas hidrelétricas e subestações elétricas. São mostrados, também, determinados aperfeiçoamentos recentes, como conversão em vídeo, telas de alarmes sonoros e visuais, ambientação sonora em três dimensões e narração do processo. Além da apresentação do Sistema ITV, são expostos o dispositivo Kinect e o algoritmo utilizado na comparação dos padrões de movimento, o DTW. Em seguida, são abordados em detalhes o projeto e a implementação da navegação, do reconhecimento de gestos e do controle de interface. Como estudo de caso, é exibida uma Instrução Técnica Virtual (ITV), elaborada especialmente para testar e avaliar a nova interface proposta. Posteriormente, são apresentados os resultados, considerados satisfatórios, obtidos através da análise de questionários qualitativos aplicados a estudantes da Universidade Federal do Pará. Por fim, são realizadas as considerações referentes a este trabalho e expostas idéias de trabalhos futuros.

PALAVRAS-CHAVES: Realidade Virtual, Treinamento Virtual, Reconhecimento de Gestos, Kinect

ABSTRACT

This work presents the implementation of navigation in the virtual environment, gesture recognition and interface control, able through the Kinect device, in the ITV System: a system for training operators and maintainers of hydroelectrics and electrical substations. In addition, it shows some recent improvements, such as conversion of a ITV on a video file, audio and visual alarms, ambience sound in three dimensions and narration of the process. Besides presenting the ITV System, are exposed to the Kinect device and the algorithm used to compare patterns of movement, the DTW. Then, this work explains in detail the design and the implementation of the navigation, the interface control and the gesture recognition. As a case study, it displays a Virtual Technical Instruction (ITV), designed especially to test and evaluate the new proposed interface. Subsequently, this work presents satisfactory results, obtained through the analysis of qualitative questionnaires applied to the students of the Federal University of Pará. Finally, considerations for this study and ideas for future work are made.

KEYWORDS: Virtual Reality, Vitrual Training, Gesture Recognition, Kinect

LISTA DE ILUSTRAÇÕES

Figura 3.1 - Manual técnico da Caixa de Mola.	23
Figura 3.2 - Foto real da Caixa de Mola.	23
Figura 3.3 - Procedimento de desmontagem no Sistema ITV.....	24
Figura 3.4 - Interface do Sistema ITV - Linhas do Tempo.	25
Figura 3.5 - Interface completa do Sistema ITV.	26
Figura 3.6 - Alerta visual e sonoro de mau funcionamento de equipamento.	28
Figura 3.7 - <i>Pop-up</i> com narração em áudio.	29
Figura 4.1 - Microsoft Kinect e seus componentes.	30
Figura 4.2 - Orientação dos eixos do Kinect.	31
Figura 4.3 - Interação de <i>hardware</i> e <i>software</i> com a aplicação.	33
Figura 4.4 - Esqueleto rastreado pela câmera RGB.	33
Figura 4.5 - Esqueleto rastreado pela câmera de profundidade.....	34
Figura 4.6 - Articulações detectadas pelo Kinect.	34
Figura 4.7 - Distância de rastreamento do Kinect.	35
Figura 4.8 - Kinect pode reconhecer até seis pessoas e rastrear duas.....	36
Figura 4.9 - Comparações de curvas entre a distância Euclidiana e DTW.	37
Figura 4.10 - Exemplo de comparação de duas curvas por DTW.	38
Figura 5.1 - Gráfico dos dados da mão	43
Figura 5.2 - Gráfico dos dados do pulso	44
Figura 5.3 - Gráfico dos dados do cotovelo.	44
Figura 5.4 - Visão da câmera no rastreamento da posição do mouse.	48
Figura 5.5 - Algoritmo de controle do mouse.	49
Figura 5.6 - Algoritmo utilizado na função de clique do mouse.	50
Figura 5.7 - Movimentos para translação de câmera no Sistema ITV.....	51

Figura 5.8 - Região neutra de posicionamento da câmera.	52
Figura 5.9 - Movimentos para a rotação de câmera no Sistema ITV.	53
Figura 5.10 – Regiões de segmentadas para rotação da câmera.....	54
Figura 5.11 - Possibilidades de movimentação da câmera.....	55
Figura 5.12 – Ícone do Kinect (a) inativo e (b) ativo.....	58
Figura 5.13 – Caixa de diálogo da configuração do passo.....	59
Figura 5.14 - Caixa de diálogo da lista de movimentos para transição.	59
Figura 5.15 - Diagrama de atividades.	61
Figura 6.1 – Navegação no ambiente através do Kinect.....	62
Figura 6.2 - Passo com transição Condicional de seleção da chave de fenda.	63
Figura 6.3 – Controle de interface para a seleção da chave de fenda.	63
Figura 6.4 - Passo com transição de Movimento de chave de fenda.	64
Figura 6.5 – Simulação de movimento de chave de fenda.....	64
Figura 6.6 – Passo com transição Condicional de seleção de martelo.....	65
Figura 6.7 - Controle de interface para a seleção do martelo.....	65
Figura 6.8 – Passo com transição de Movimento de martelo.....	66
Figura 6.9 - Simulação de movimento de martelo.....	66
Figura 6.10 – Passo com transição Condicional de seleção de chave combinada.....	67
Figura 6.11 – Controle de interface para a seleção da chave combinada.	67
Figura 6.12 – Passo com transição de Movimento de chave de boca.....	68
Figura 6.13 – Simulação de movimento de chave de boca.	68
Figura 7.1 – Grau de familiaridade com o Sistema ITV.	70
Figura 7.2 – Satisfação sobre o controle de interface através do Kinect.	71
Figura 7.3 – Satisfação sobre a navegação no ambiente através do Kinect.	72
Figura 7.4 – Resultados sobre o Kinect em relação ao mouse e teclado.	73
Figura 7.5 – Satisfação sobre o reconhecimento de gestos através do Kinect.	73

Figura 7.6 – Dificuldade no reconhecimento dos gestos. 74

Figura 7.7 – Grau de atratividade do aprendizado através do Kinect..... 75

LISTA DE TABELAS

Tabela 4.1 - Especificação dos componentes do Kinect.	32
Tabela 5.1 - Saída do DTW para movimento de martelo na articulação da mão.	41
Tabela 5.2 - Limiares definidos.	42
Tabela 5.3 - Teste da quantidade de valores por padrão.	43
Tabela 5.4 - Resultado do teste de hierarquia.	46
Tabela 5.5 - Resultado da validação.	46
Tabela 5.6 - Equivalência entre funções e interfaces de entrada.	55

LISTA DE EQUAÇÕES

Equação 5.1 - Vetor original.	40
Equação 5.2 - Vetor normalizado.	40
Equação 5.3 - Filtro passa-baixas.	49

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVOS	14
1.1.1	Objetivo Geral	14
1.1.2	Objetivos Específicos	14
1.2	COMPOSIÇÃO	14
2	TRABALHOS CORRELATOS	16
2.1	TREINAMENTO VIRTUAL	16
2.1.1	Subestações Elétricas	17
2.1.2	Utilizando NUI's	18
2.2	KINECT COM DTW	19
3	SISTEMA ITV	22
3.1	CONSTRUÇÃO DE UMA ITV	22
3.2	APERFEIÇOAMENTOS RECENTES	27
4	KINECT E DTW	30
4.1	KINECT	30
4.2	DYNAMIC TIME WARPING	36
5	PROJETO E IMPLEMENTAÇÃO	39
5.1	RECONHECIMENTO DE GESTOS	39
5.1.1	Normalização	39
5.1.2	Detecção de Movimentos	40
5.1.3	Hierarquia de Reconhecimento	45
5.2	CONTROLE DA INTERFACE	46
5.2.1	Cursor do Mouse	47
5.2.2	Clique do Mouse	50

5.3	NAVEGAÇÃO NO AMBIENTE.....	51
5.4	INTEGRAÇÃO DO KINECT AO SISTEMA ITV.....	56
5.4.1	Classes e Bibliotecas	56
5.4.2	Interface Gráfica	57
5.4.3	Funcionamento	59
6	ESTUDO DE CASO	62
6.1	ITV DE TESTE E AVALIAÇÃO	62
7	AVALIAÇÃO	70
7.1	RESULTADOS	70
8	CONCLUSÃO E TRABALHOS FUTUROS.....	76
	REFERÊNCIAS.....	78
	APÊNDICE A - QUESTIONÁRIO DE SATISFAÇÃO	84

1 INTRODUÇÃO

Nos últimos anos a interação com os computadores através de gestos tem movimentado a área industrial e o mercado de jogos devido aos recentes avanços nos algoritmos e na tecnologia. Estes avanços permitem rastrear o corpo humano e identificar gestos e movimentos, tornando o controle da máquina uma atividade mais simples e natural (NORMAM, 2010). Esta forma de interagir com computadores é denominada *Natural User Interface* (NUI), que segundo Jain, Lund e Wixon (2011), é a forma que permite que os usuários interajam com os computadores da maneira que interagem com o mundo.

Devido a sua simplicidade, interfaces gráficas são mais bem sucedidas que interfaces de linhas de comando. No mesmo pensamento, interfaces naturais devem ser mais simples para serem bem aceitas. O nível de simplicidade de uma interface está relacionado com a facilidade que o usuário consegue lembrar do movimento correto para executar alguma ação e, assim, tornar o controle do sistema algo natural (NORMAM, 2008).

A Realidade Virtual (RV) é a representação do mundo real em ambientes virtuais, geralmente tridimensionais, ou seja, permite que pessoas interajam com ambientes conceituais a partir do mundo real (KIRNER e TORI, 2004). Os ambientes e técnicas de RV têm sido empregados em diversas áreas da computação, como o tratamento de fobias, entretenimento, demonstração de ambientes, capacitação de pessoas e outros (TORI, KIRNER e SISCOUTO, 2006).

O treinamento é uma forma específica de aquisição do conhecimento que, na maioria das vezes, torna-se um processo caro e envolve riscos físicos aos aprendizes. O aprendizado utilizando RV possibilita que o profissional repita a atividade e analise seu desenvolvimento, construindo uma base sólida de conhecimento de forma menos arriscada (SILVA, MIRANDA e HOUNSELL, 2007). Devido a essas vantagens, os autores Lopes, Oliveira e Silva (2010) consideram o uso dos simuladores de realidade virtual como a solução mais eficiente de treinamento, possibilitando capacitar pessoas de maneira rápida e com baixo custo.

O Sistema ITV é um *software* de treinamento virtual desenvolvido pelo Laboratório de Realidade Virtual (LARV) da Universidade Federal do Pará (UFPA), com financiamento da empresa pública Eletronorte - Centrais Elétricas do Norte do Brasil S/A, utilizado para simular, com um grau elevado de detalhes, instruções técnicas de operação (ITO) e instruções técnicas de manutenção (ITM) de uma usina hidrelétrica e de uma subestação elétrica. Este

sistema possui um modo de autoria, onde o usuário pode criar ou modificar os denominados “passos” dessas instruções técnicas. Assim, o usuário pode consertar, atualizar ou até mesmo criar uma nova e completa instrução técnica sem a necessidade de qualquer conhecimento em programação (JUNIOR, 2006) (JUNIOR *et al*, 2009).

O produto final da utilização do sistema é uma Instrução Técnica Virtual (ITV), que é a transformação da descrição textual de uma ITM ou ITO em uma aplicação visual, auditiva e interativa do processo, tornando o aprendizado mais rápido e atraente. Desta forma, atende aos requisitos principais de treinamento virtual (BLUEMEL *et al*, 2003) (BLUEMEL *et al*, 2005).

A forma de interação atual do Sistema ITV é feita por meio de mouse e teclado, onde deve-se clicar em determinados locais na tela do ambiente 3D para que haja continuidade no procedimento. Para indivíduos que possuem familiaridade com computadores, estes dois dispositivos podem ser fáceis de utilizar, porém, para os que não possuem, esta forma de interação pode não se assemelhar aos procedimentos e condições reais, logo, dificultando o aprendizado.

Portanto, com o objetivo de conferir um maior grau de realidade e imersão no ambiente virtual, o foco deste trabalho está na implementação de uma nova forma de interação com o sistema, um dispositivo de Interação Homem-Máquina (HCI) (HELANDER, LANDAUER e PRABHU, 1997).

Este dispositivo de HCI, denominado Kinect, foi desenvolvido pela empresa Microsoft e possibilita o reconhecimento e rastreamento do corpo humano sem a utilização de nenhum aparelho ou marcador junto ao corpo (KINECT, 2012). Dessa maneira, permite que operadores e mantenedores executem movimentos similares aos procedimentos durante o treinamento, conferindo um maior grau de realidade por melhorar a capacidade de imersão no sistema e promovendo uma interface natural para navegação e interação com o ambiente virtual (BARTOLI *et al*, 2012). Essas vantagens na interação com o ambiente levaram o Kinect a ser escolhido para ser testado e avaliado no Sistema ITV.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Integrar o dispositivo Kinect ao Sistema ITV a fim de melhorar a forma de interação entre o usuário e o sistema.

1.1.2 Objetivos Específicos

- Desenvolver a navegação no ambiente virtual através de movimentos com as mãos
- Reconhecer gestos pré-definidos que simulam a utilização de ferramentas
- Controlar a interface por meio da movimentação das mãos
- Analisar a aceitação da nova interface proposta através de questionários respondidos por estudantes da UFPA

1.2 COMPOSIÇÃO

Além deste capítulo de introdução que descreveu resumidamente o objetivo do trabalho e a motivação de sua realização, existem ainda 7 capítulos.

O capítulo 2 apresenta trabalhos correlatos de treinamento virtual, a utilização de NUI's em treinamentos e o dispositivo Kinect trabalhando junto ao DTW. Serão analisadas as vantagens e desvantagens a título de comparação com o Sistema ITV ao fim desta dissertação.

O capítulo 3 discorre sobre o Sistema ITV, onde serão detalhadas algumas características e funcionalidades necessárias para o entendimento do trabalho, bem como melhorias recentes que conferem maior grau realístico ao sistema.

O capítulo 4 explica o dispositivo Kinect, seus componentes, suas vantagens e desvantagens, o modo como reconhece os movimentos, dentre outros aspectos. Além disso, neste capítulo é mostrado o funcionamento do algoritmo *Dynamic Time Warping*, suas definições teóricas e propriedades matemáticas.

No capítulo 5 tem-se o projeto e implementação do Kinect no Sistema ITV, bem como a metodologia aplicada para o reconhecimento de gestos e o funcionamento do controle da interface e da navegação no ambiente virtual.

O capítulo 6 exhibe, como estudo de caso, uma instrução técnica virtual elaborada especialmente para teste e avaliação da NUI integrada ao sistema.

O capítulo 7 apresenta a avaliação e comenta os resultados obtidos através de questionários qualitativos aplicados a estudantes da UFPA.

No capítulo 8 são tecidas as conclusões e expostas idéias de trabalhos futuros para o Sistema ITV.

2 TRABALHOS CORRELATOS

Neste capítulo serão mostrados alguns trabalhos de treinamento virtual, com e sem a utilização do Kinect, sendo que em alguns desses trabalhos utiliza-se a metodologia DTW para reconhecimento de gestos.

2.1 TREINAMENTO VIRTUAL

No artigo de Lopes *et al* (2008), os autores avaliam o treinamento de operadores de *harvester* (tipo de veículo pesado para o corte de árvores) com uso de simulador de realidade virtual. Este trabalho foi conduzido no Centro de Formação de Operadores Florestais (CENFOR) do Departamento de Engenharia Florestal da Universidade Estadual do Centro-Oeste (UNICENTRO), em Irati, Paraná. A população pesquisada foi composta por 39 trabalhadores de diversas empresas florestais, sem experiência em *harvester*, que receberam treinamento, com carga horária de 40h, em simulador de realidade virtual. Foram avaliados diferentes módulos operacionais e os resultados obtidos pelos autores indicaram que o simulador: reduziu o tempo de aprendizado, sendo o melhor desempenho adquirido no início do treinamento, melhorou significativamente o desempenho dos operadores no decorrer do treinamento e mostrou que a carga horária de 40h foi suficiente para a obtenção de resultados satisfatórios. O melhor desempenho dos operadores ocorreu com a variável produtividade, cujo ganho médio foi de 41,3% em relação ao início do treinamento, seguido pelo tempo de execução, com 38,8%; e pela direção de queda, com ganho médio de 36,1%. Concluem, então, que o simulador mostrou ser uma ferramenta de treinamento de grande eficiência no aprendizado e na formação do operador, com vantagens no menor custo de treinamento e na redução do tempo de aprendizado.

Na área da medicina, o trabalho desenvolvido por Lima *et al* (2004) apresenta o protótipo de uma ferramenta baseada em características de RV para contribuir com o treinamento de procedimentos cirúrgicos, através da implementação de um protótipo de ferramenta para simular o exame de punção aspirativa da mama. Este sistema foi implementado em Java utilizando a *Application Programming Interface* (API) Java 3D (JAVA, 2012), tendo o MySQL (MYSQL, 2012) como suporte para o banco de dados e imagens e o aplicativo 3DS Max (MURDOCK, 2007)(CLINTON, 2007) para a modelagem dos objetos 3D utilizados no protótipo.

Foram modelados tridimensionalmente os seios e as seringas utilizadas no processo de punção e foi criado um banco de dados que manipulavam dados dos pacientes e das imagens disponibilizadas nos casos incluídos na ferramenta. O treinamento consistia em analisar os dados do paciente e localizar, nas imagens, o nódulo e onde a agulha deveria penetrar para efetuar a punção aspirativa. Os autores concluem que o protótipo pode oferecer a prática aos médicos recém-formados e, como melhora, sugeriram a utilização de dispositivos hápticos, ou seja, dispositivos que possam simular a posição e a pressão da punção, entre outras características.

2.1.1 Subestações Elétricas

Na área da engenharia, mais especificamente em subestações elétricas, o trabalho Quintana e Mendonza (2009) apresenta uma metodologia que tem sido desenvolvida e implementada com sucesso na maneira de criar modelos virtuais completos para subestações. Os autores descrevem diferentes estágios no processo de modelagem, mostrando como têm enfrentados os problemas para representar objetos reais. Também expõem fatores necessários na construção dos modelos e mostra uma aplicação sobre a subestação de Xochimilco, no México. Essa subestação tem a intenção de ser expandida, mas há dúvidas sobre a viabilidade do projeto devido ao espaço disponível. Sendo este um dos motivos para ser construída virtualmente.

Por fim, concluem que os modelos tridimensionais possibilitam uma melhor perspectiva sobre os projetos, até antes de serem executados, e dá a chance de testar várias possibilidades com alto grau de realismo.

Outro artigo da área, desenvolvido por Arroyo e Arcos (1999), traz uma aplicação denominada *Power Distribution Application* (PDA), baseada em RV do tipo desktop, para treinamento de pessoas associadas à operação manual de equipamentos dessas subestações. O objetivo do PDA é melhorar o modo guiado do *Silicon Graphics International* (SGI) *Simulator*, um sistema amplamente usado pela empresa espanhola de geração e distribuição de energia elétrica Iberdrola (IBERDROLA, 2012) para o treinamento de operadores na execução de manobras locais nas subestações onde não existe automação ou comando remoto. O PDA é composto por dois sistemas independentes: um simulador de plantas (SGI Simulator) e o Assistente. Este último controla e supervisiona a execução dos exercícios do usuário de duas formas: reativa e pró-ativa. A forma reativa consiste na geração de

explicações quando o usuário comete um erro e a pró-ativa consiste em explicações sobre diferentes aspectos de manobras e equipamentos elétricos.

O sistema proposto possui três funções principais que permitem que o operador aprenda: (1) como ver a aparência, topologia e localização física dos equipamentos elétricos, (2) como operar local e remotamente disjuntores, interruptores, dentre outros e (3) como operar a subestação e ver seus efeitos no SGI Simulator a fim de seguir corretamente os procedimentos de operação estabelecidos. Os autores concluem que o desenvolvimento do PDA demonstrou ser viável para a empresa Iberdrola devido ao seu baixo custo, utilizando a arquitetura de RV do tipo desktop.

2.1.2 Utilizando NUI's

Em seu trabalho, Fonseca (2013) faz uso do Sistema ITV para implementar o controle de interface, navegação e reconhecimento de gestos utilizando um dispositivo criado pela empresa Nintendo, denominado *Wii Remote Controller* (NINTENDO, 2013). A proposta do trabalho se assemelha à desta dissertação e o intuito futuro é comparar os dois dispositivos para verificar quais vantagens e desvantagens de ambos.

O *Wii Remote Controller* é um dispositivo com comunicação sem fio (*Bluetooth*) que possui diversos botões, *led's*, câmera acoplada, *rumble* (vibrador), *speaker* (caixa de som), memória interna e acelerômetros. O botões, *led's*, *rumble*, *speaker* podem ser configurados para exercer as funcionalidades que o desenvolvedor necessite. A memória interna pode ser utilizada para gravar dados personalizados e os acelerômetros fornecem as informações de translação e rotação, necessárias para o reconhecimento de gestos. A câmera acoplada, utilizada em conjunto com uma barra emissora de infravermelhos, pode detectar a posição do controle em relação ao televisor ou monitor, ideal para controlar o cursor na tela.

Em Fonseca (2013), o algoritmo utilizado no reconhecimento dos gestos é o *Dynamic Time Warping* e os gestos a serem reconhecidos foram o movimento de martelo e dois movimentos de chave de fenda: sentido horário e anti-horário. Foi utilizado um conjunto com 20 amostras de cada movimento e, destas, foram tiradas a média, assim, formou-se uma sequência de valores padrão para cada gesto. Por fim, foi criado um conjunto com outras 25 amostras para teste. O resultado obtido foi de 84% de acerto no movimento de martelo, 72% no movimento horário e 92% no movimento anti-horário. O *Wii Remote Controller* e o algoritmo DTW foram implementados no Sistema ITV e o resultado final obtido é mostrado

com uma ITV criada especialmente para a ocasião, onde a aceitação foi considerada satisfatória.

No artigo de Bartoli *et al* (2012), é apresentado um protótipo de um sistema para simulação e treinamento virtual de médicos e paramédicos em situações de emergência, de um caso mais comum, como uma pessoa inconsciente no chão, até o extremo, como um acidente de carro com algumas pessoas machucadas. Os autores iniciam o artigo frisando a importância do treinamento em situações emergenciais, onde a prática é a chave para a manutenção e o aprendizado das habilidades na medicina. Em seguida, citam importantes características da categoria de treinamento médico virtual, como a possibilidade de repetir diversos cenários complexos a um preço baixo, sem que cause risco para os médicos.

O protótipo desenvolvido simula virtualmente os atendimentos necessários para os primeiros socorros em um ambiente gráfico 3D. Também há um criador de cenários, que permite que os treinadores definam situações complexas e variadas para manter os alunos focados no ambiente, na condição do paciente e nos monitores. Os usuários interagem com o reconhecedor de gestos Kinect, que melhora a capacidade de imersão do sistema e provê uma interface natural de navegação e interação com o ambiente virtual.

Em seguida, os autores discutem alguns cenários e estudos de caso, mostram como são a navegação e a interação no sistema e fazem um breve resumo de como foi a aceitação dos médicos, usuários do aplicativo. Concluem, então, que a simulação na medicina tem melhorado bastante na fidelidade com a realidade e com a validade do treinamento, contudo, ainda existem algumas barreiras que justificam o ceticismo na adoção do sistema como um processo de treinamento. A maior razão desse atraso é o custo e a logística para criação de uma larga estrutura que necessita de colaboração entre diferentes unidades de hospitais.

2.2 KINECT COM DTW

O artigo de Gonçalves *et al* (2012) apresenta o desenvolvimento de uma metodologia para detectar automaticamente movimentos estereotipados de indivíduos com desordem do espectro autista (ASD). Neste trabalho é utilizado o sensor Microsoft Kinect e o algoritmo DTW para o reconhecimento de gestos.

Os autores introduzem o trabalho explicando o método tradicional para a detecção de ASD, que é feito com escalas de avaliação em papel e caneta, onde se observa repetidos movimentos. Em seguida, explicam que os sistemas desenvolvidos recentemente para detectar

comportamentos estereotipados têm utilizado acelerômetros com algoritmos de identificação de padrões, porém, para esse tipo de dispositivo, é necessário que braçadeiras sejam colocadas no pulso. Crianças com autismo podem resistir a este tipo de situação. Desta forma, o Kinect pode suprimir essas desvantagens, visto que não é preciso carregar nenhum dispositivo ou marcador junto ao corpo.

Durante o desenvolvimento do trabalho, são explicadas as vantagens e detalhes sobre o algoritmo utilizado e como foi implementado. Em seguida, os autores explicam que o sistema foi testado com cinco crianças com ASD, com média de idade de 9 anos. Por fim, concluem que o sensor Kinect obteve um resultado positivo de 83% para ASD, mas embora os resultados sejam promissores, alguns aspectos devem ser melhorados. Um destes aspectos é o ruído nas imagens de profundidade obtido pelo Kinect, que pode levar a falso-positivos.

No trabalho de Doliotis *et al* (2011) é proposto um método robusto de detecção de gestos com as mãos através do Microsoft Kinect. Os autores explicam que a interação do humano com o computador deve ser tão intuitiva e natural quanto possível, assim, o usuário deve interagir com a máquina sem o uso de marcadores ou aparatos como luvas rastreadoras, controles remotos, mouses e teclados. Em seguida, comentam que com a tecnologia do Kinect, o reconhecimento de gestos está se tornando maduro o suficiente para ser amplamente utilizado pelo público. Este método de rastreamento de mãos utiliza a detecção de movimentos baseado na diferenciação de quadros e segmentação de profundidade. Os valores de profundidade de cada pixel da imagem são obtidos pelo Kinect e submetidos ao algoritmo DTW.

Os autores descrevem o experimento feito, que definiu 10 classes representativas de dez dígitos, do 0 ao 9. Para cada dígito existem vários exemplos para treinamento na base de dados. Mais especificamente, foram 10 diferentes usuários executando 3 amostras para cada dígito, ou seja, são 300 amostras de treinamento no total.

No desenvolvimento do artigo, os autores discorrem sobre o processo de implementação da detecção dos movimentos e sobre a normalização feita para que todas as amostras e os modelos a serem comparados estivessem em uma mesma escala numérica. Em seguida, descrevem detalhadamente o algoritmo *Dynamic Time Warping*, sua lógica e suas condições características. Por fim, apresentam os experimentos realizados e os resultados obtidos, onde o método proposto pelos autores teve um índice médio de acerto de 95% no conjunto de testes. Concluem, então, que algumas restrições permanecem, por exemplo, a não variação do ponto de vista, ou seja, o método só é efetivo se o usuário estiver de frente para o

sensor. Além disso, é preciso investigar soluções para a segmentação temporal, pois o tempo de reconhecimento é pré-definido e, assim, a automatização deste processo é um trabalho futuro.

3 SISTEMA ITV

Segundo a definição de Chiavenato (1985): “Treinamento é o processo educacional, aplicado de maneira sistemática e organizada, através do qual as pessoas aprendem conhecimentos, atitudes e habilidades em função de objetivos definidos”. Este processo educacional proporciona benefícios quanto ao mercado de trabalho, quanto ao pessoal em serviço e quanto à empresa como um todo (MILKOVICH, 2000). Desta forma, o treinamento é de fundamental importância para o sucesso de uma organização, sendo “uma experiência que produz uma mudança permanente em um indivíduo e que melhora a sua capacidade de desempenhar um determinado cargo” (VOLPE, 2009).

Por essas razões, o Sistema de Autoria ITV foi desenvolvido para que qualquer procedimento possa ser criado, modificado e repetido de forma a possibilitar o aprendizado do usuário. Para as empresas do setor elétrico e até de outras áreas, este software pode ser de grande importância, uma vez que possibilita que os operadores e mantenedores transformem a antiga instrução técnica em papel para o novo modo sugerido, ou seja, convertam as apostilas de instruções técnicas em arquivos digitais contendo essas simulações.

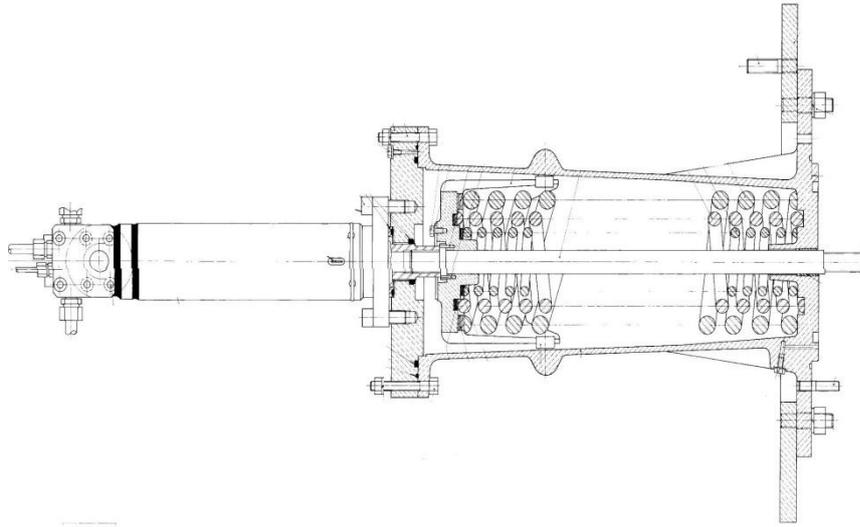
Este software foi desenvolvido utilizando recursos livres com o objetivo de redução dos custos de produção, onde a linguagem de programação adotada foi o C++ (DEITEL e DEITEL, 2009), auxiliada pelo motor gráfico Ogre3D (JUNKER, 2006) e pelo framework de interface Qt (BLANCHETTE e SUMMERFIELD, 2008).

3.1 CONSTRUÇÃO DE UMA ITV

Uma instrução técnica virtual (ITV) é o produto construído a partir do Sistema de Autoria ITV. A criação de uma ITV pode ser dividida em três etapas. Primeiramente, é realizado um levantamento de dados sobre os aspectos que cercam a instrução técnica, como o ambiente onde será feita a manutenção ou a operação, as peças que serão manipuladas, as ferramentas utilizadas, a quantidade de mantenedores e os equipamentos de proteção utilizados. Essa coleta de informações é feita através de materiais técnicos, como: manuais dos fabricantes, plantas baixas da instalação, desenhos técnicos digitais, fotos do local onde o procedimento é realizado e vídeos desses procedimentos que contenham informações úteis para a realização da instrução técnica. A Figura 3.1 mostra o manual técnico utilizado como

referência para a modelagem 3D da Caixa de Mola e a Figura 3.2 mostra uma imagem real da Caixa de Mola, obtida durante o procedimento de desmontagem.

Figura 3.1 - Manual técnico da Caixa de Mola.



Fonte: ELETRONORTE, 2012.

Figura 3.2 - Foto real da Caixa de Mola.



Fonte: Próprio autor.

De posse dos dados necessários, ou seja, desenhos técnicos em *computer-aided design* (CAD), fotos, vídeos e manuais de procedimento, inicia-se a segunda etapa para a criação da ITV: a modelagem tridimensional. Assim, é feita a modelagem 3D de todas as peças a serem

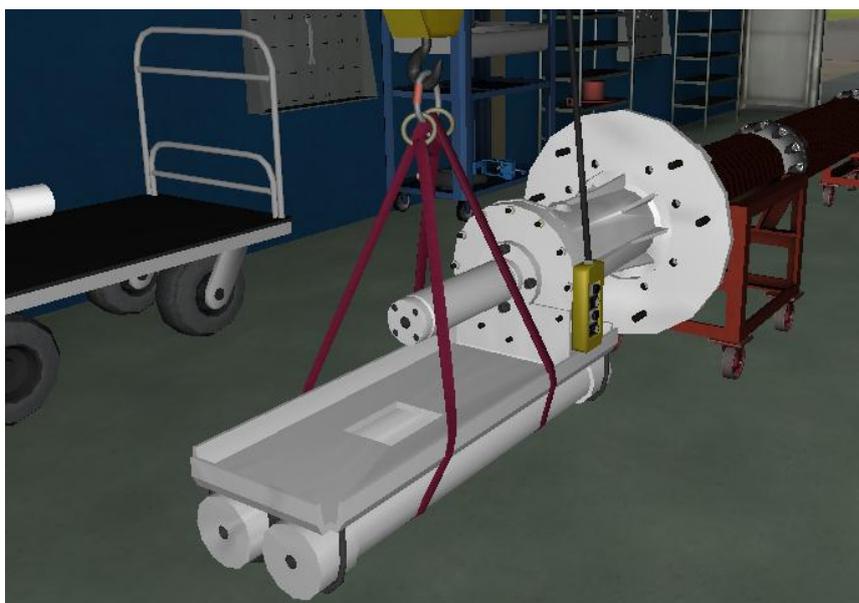
utilizadas na ITV, onde o *software* empregado foi o 3DSMAX 9 (MURDOCK, 2007)(CLINTON, 2007), acrescido do *plugin* oFusion para exportação dos modelos no formato Mesh (OFUSION, 2012).

Após a modelagem 3D da peça, é criada a textura através de fotos reais do objeto e tratada graficamente com o auxílio do *software GNU Image Manipulation Program* (GIMP, 2012). Nesse processo, são utilizadas diversas fotos para que os diferentes revestimentos de peças, parafusos ou porcas sejam obtidos. Em seguida, é feito o mapeamento de cada polígono (ou face) da peça 3D em seu respectivo lugar na textura gerada. Este processo é repetido para todas as peças a serem utilizadas em uma ITV.

Depois do processo de modelagem, tratamento e texturização concluídos, esses objetos são exportados para arquivos no formato reconhecido pelo Ogre3D, o Mesh. Finalmente, com os objetos necessários para a criação da ITV o usuário pode carregar todas as peças no Sistema ITV e iniciar o desenvolvimento de uma instrução técnica seguindo o manual de procedimento, sendo esta a terceira e última etapa.

Normalmente, em uma instrução técnica de manutenção (ITM) virtual, são carregadas apenas as peças e ferramentas que serão manipuladas e o ambiente onde a manutenção irá ocorrer, por exemplo, uma oficina. A Figura 3.3 ilustra uma etapa do procedimento de desmontagem da Caixa de Mola no Sistema ITV.

Figura 3.3 - Procedimento de desmontagem no Sistema ITV.



Fonte: Próprio autor.

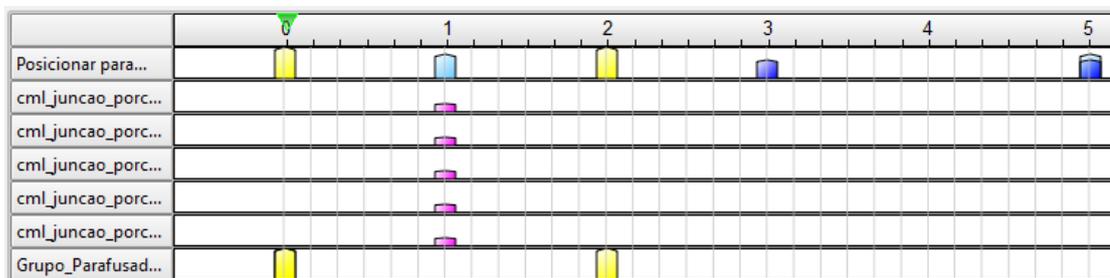
Essa redução na quantidade de peças carregadas ocorre devido à preocupação com a sobrecarga de processamento e memória. Já numa instrução técnica de operação (ITO) virtual, normalmente, todo o ambiente da subestação elétrica é carregado, contudo, os objetos modelados são feitos com menos detalhes para evitar a sobrecarga e somente objetos que serão o foco da ITO são detalhados. Desta forma, há um compromisso entre detalhamento de peças *versus* sobrecarga do sistema, que precisa ser avaliado e obedecido para o bom funcionamento da ITV.

Deste ponto em diante, a construção da ITV é feita visualmente através de uma sequência de passos e cada passo representa uma etapa no procedimento de manutenção descrito na Instrução Técnica.

Cada passo contém uma Linha do Tempo para cada peça possa sofrer transformações como: rotação, translação, escala ou mesmo o som da peça. As linhas do tempo apresentam marcadores que indicam o instante inicial e final de execução das transformações de cada peça no decorrer do passo. A distinção entre as transformações são feitas através das cores dos marcadores, ou seja, cada transformação da peça possui um marcador na sua linha do tempo e, dependendo da transformação configurada, o marcador se estabelece em uma cor diferente.

Desta forma, através da linha do tempo o construtor da ITV pode definir quais transformações cada peça sofrerá e o seu tempo de duração. A Figura 3.4 mostra um exemplo das linhas do tempo de algumas peças.

Figura 3.4 - Interface do Sistema ITV - Linhas do Tempo.



Fonte: Próprio autor.

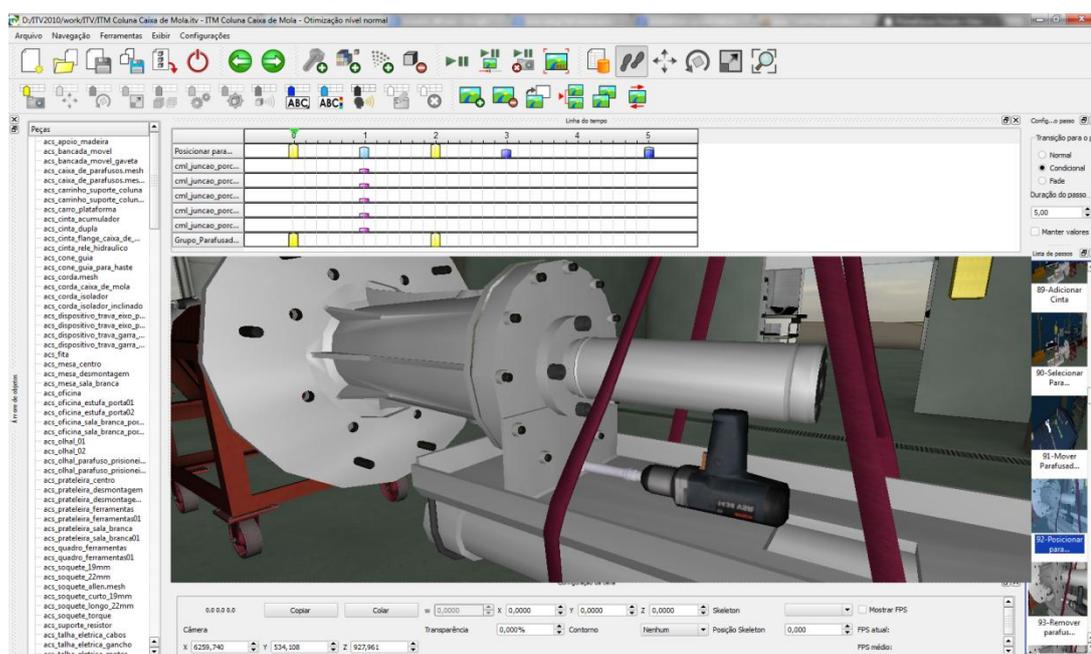
O construtor da ITV pode também criar avisos *pop-ups* que aparecem durante a execução. Esses *pop-ups* podem servir para narrar o procedimento, explicar o funcionamento

da peça, alertar sobre algum possível engano, dentre outras funções. Esses avisos são configurados através de marcadores na linha do tempo.

Para simular um treinamento, o usuário precisa interagir com o sistema. Essa interação pode ser feita facilmente pelo construtor da ITV, bastando selecionar o objeto de interação para que o procedimento continue sendo executado corretamente e configurar esta função a ele. Normalmente, cria-se um *pop-up* auxiliar para que o usuário saiba que o sistema está esperando uma interação e, às vezes, para ajudá-lo durante o procedimento. Por vezes, durante a simulação de instruções técnicas, algumas peças merecem destaque ou foco em determinadas situações. Além dos avisos na tela, é possível também dar esse destaque individualmente a cada objeto na cena, que irá aumentar e diminuir a intensidade de seu brilho, repetidamente, na cor configurada para aquela ação. Esse artifício pode garantir um melhor entendimento para usuários que estão se habituando aos nomes das peças ou pode servir para explicar situações perigosas, como altas temperaturas ou pressões.

Por fim, após todas as peças carregadas e todos os passos criados com suas devidas modificações, o construtor da ITV pode salvar um arquivo com a extensão .itv, que contém todas as informações necessárias para o treinamento em qualquer computador que contenha o Sistema ITV. A Figura 3.5 mostra a interface do Sistema ITV, no modo de autoria com todas as funcionalidades.

Figura 3.5 - Interface completa do Sistema ITV.



Fonte: Próprio Autor.

Pode ser observado que no menu superior estão as opções de modificações de cenas e peças como: adição de *pop-ups*, rotação, movimentação e escala. Na parte lateral esquerda tem-se a lista de peças carregadas, onde é possível selecionar uma ou várias peças para que sejam aplicadas as condições necessárias. No menu lateral direito está a lista de passos que formam a seqüência do procedimento e podem ser manipulados livremente. Acima da lista de passos está a configuração de transição de passo, tendo como opções: Normal, *Fade* e Condicional. No menu inferior é possível observar as configurações de câmera e os parâmetros de configuração das modificações realizadas nas peças. Por fim, no centro da interface tem-se a cena renderizada como resultado de todo o processo e, logo acima, a linha do tempo já citada.

3.2 APERFEIÇOAMENTOS RECENTES

Para a reprodução de um arquivo que contém uma ITV, é preciso que o computador tenha o Sistema ITV instalado, porém, nem sempre isso é possível. Além disso, é necessário que o computador possua uma placa gráfica com memória dedicada e um processador atualizado para que o *software* possa ser executado sem dificuldades. A solução encontrada, sugerida por Barata (2010), foi transformar a simulação de todo o procedimento da ITV, com todas suas características gráficas, em vídeo, onde o formato é o *Audio Video Interleave* (AVI). Este formato é atualmente um dos mais utilizados na reprodução de vídeos em computadores e outros dispositivos reprodutores de mídia.

Esta solução foi desenvolvida através da inserção do FFmpeg, uma biblioteca completa, livre e multiplataforma, para conversão de áudio e vídeo (FFMPEG, 2013). Dentro do Sistema ITV, essa API tem como finalidade a captura e armazenamento da cena, em uma quantidade pré-configurada de quadros por segundo e, em seguida, a conversão dos quadros armazenados em um arquivo de vídeo. Com essa melhoria, torna-se possível que uma ITV seja reproduzida em qualquer computador ou até mesmo em outros dispositivos como DVD's, celulares e *tablets*, possibilitando, inclusive, a reprodução de instruções técnicas virtuais nos trabalhos em campo.

Foram realizados, também, alguns aperfeiçoamentos envolvendo áudio. A API utilizada para a manipulação de som foi a FMOD, uma biblioteca de programação e um

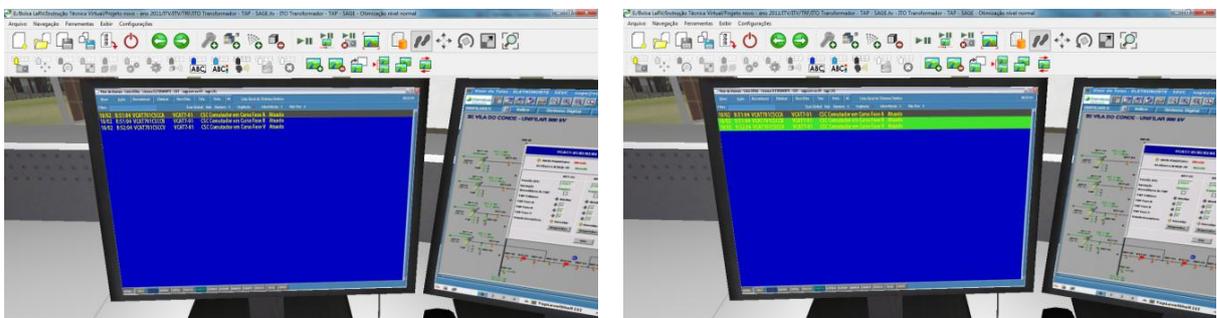
conjunto de ferramentas para a criação e reprodução de áudio interativo, que suporta os principais sistemas operacionais e plataformas de jogos (FMOD, 2012).

Uma dessas melhorias do sistema é a inserção de áudio 3D, que confere uma sensação de imersão no ambiente, pois funciona relacionando o objeto ao som. Assim, ao aproximar ou afastar a câmera do objeto, o som aumenta ou diminui, respectivamente, sua intensidade. Do mesmo modo, se este objeto está à esquerda na cena, há uma maior intensidade de som oriunda da saída de áudio esquerda. Pode-se entender o áudio 3D como uma espécie de som pontual na cena, se adequando à posição do espectador, assim, este efeito pode garantir um maior grau de realismo nos treinamentos. Esse aspecto multidisciplinar (visual e auditivo) tem sua importância destacada por Machado e Moraes (2010).

Devido o sistema permitir a interação com peças, botões e telas durante a execução de uma ITV, esta nova função de áudio permite também que sejam adicionados alarmes sonoros relacionados às telas dos operadores, ideais para simulação de erros em instruções técnicas de operação (ITO's). As telas utilizadas são representações fiéis do Sistema Aberto de Supervisão e Controle (SAGE, 2012).

Outra inovação, que complementa os sons de alarmes, é a possibilidade de animação em textura, ou seja, a modificação da textura utilizada em tempo de execução. Assim, os alarmes simulam um ambiente real de mau funcionamento de equipamentos com alarmes visuais e sonoros nas telas dos operadores. A animação de textura pode ser mais bem compreendida na Figura 3.6, que ilustra a simulação de alerta do Sistema Aberto de Supervisão e Controle (SAGE) recebido pelo operador .

Figura 3.6 - Alerta visual e sonoro de mau funcionamento de equipamento.



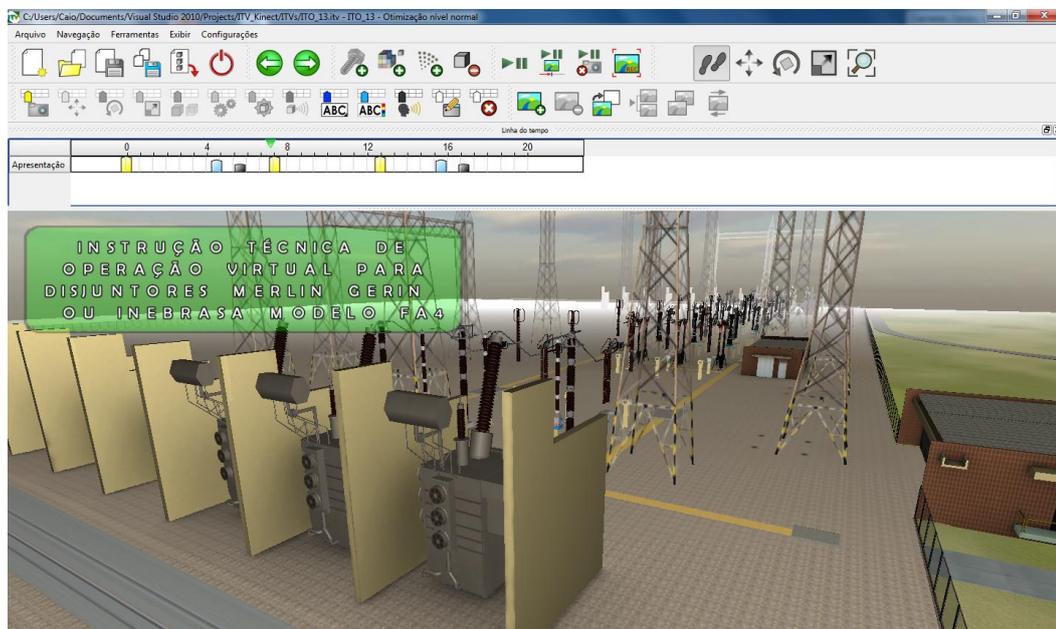
Fonte: Próprio Autor.

Esta tela de alarme tem sua textura alternada constantemente, a cada segundo, simulando de forma idêntica o alerta real do SAGE. Além disso, o alarme sonoro também entra em repetição constante até que seja tomada uma providência por parte do operador, ou seja, que o problema seja reconhecido e solucionado.

Por último, um importante aperfeiçoamento envolvendo áudio, é a possibilidade de narração do procedimento em determinadas situações. Isso facilita a absorção do conhecimento e torna o treinamento menos monótono. Além disso, o usuário pode conhecer os nomes e as pronúncias corretas das peças que estão sendo manipuladas no momento, entender o funcionamento dos componentes e até ouvir informações sobre segurança.

Para que seja narrado um *pop-up*, é preciso que o texto seja previamente gravado por uma pessoa. Em seguida, o arquivo de áudio pode ser inserido na ITV e manipulado na linha do tempo do passo através do marcador de cor preta. A Figura 3.7 mostra um exemplo de um *pop-up* com narração em áudio.

Figura 3.7 - Pop-up com narração em áudio.



Fonte: Próprio autor.

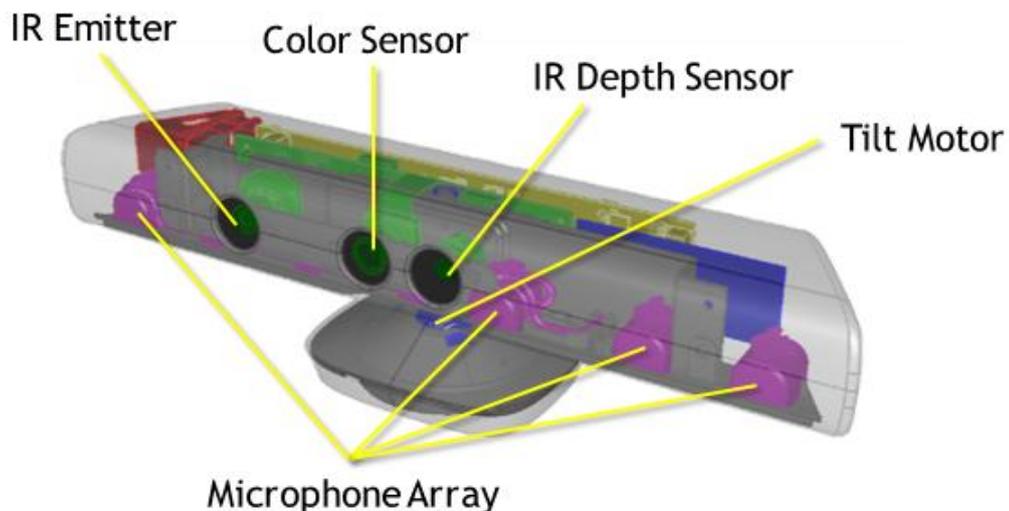
4 KINECT E DTW

Neste capítulo será apresentado o sensor Kinect, utilizado para o rastreamento e aquisição de dados do corpo do indivíduo. Em seguida, será mostrada uma introdução sobre o algoritmo *Dynamic Time Warping* (DTW), utilizado para a comparação entre padrões de movimento.

4.1 KINECT

Kinect é um dispositivo criado pela empresa Microsoft, disponível para os usuários a partir de Novembro de 2010, que possibilita rastrear e identificar o corpo humano sem o auxílio de nenhum marcador ou objeto junto à pessoa. Para isso, utiliza-se uma câmera *Red Green and Blue* (RGB), uma câmera de profundidade, um emissor de infravermelhos (IR), quatro microfones alinhados através de sua estrutura e ainda conta com um servo-motor e um acelerômetro interno (KINECT, 2012). A Figura 4.1 mostra o dispositivo Microsoft Kinect e seus componentes.

Figura 4.1 - Microsoft Kinect e seus componentes.

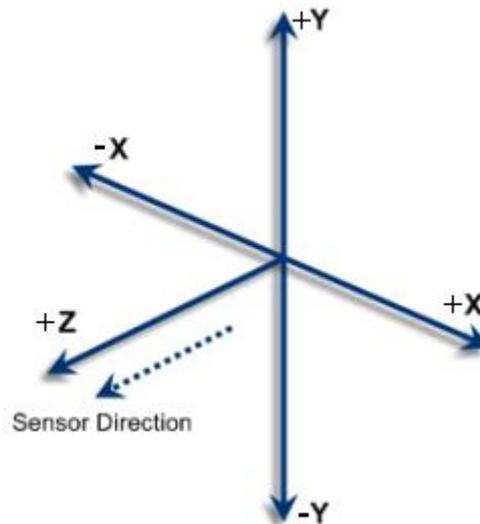


Fonte: KINECT, 2012.

A câmera RGB é responsável por fornecer as imagens coloridas e as informações do esqueleto rastreado nas dimensões vertical e horizontal (eixos Y e X, respectivamente). Já a

câmera de profundidade, que nada mais é que um sensor de infravermelhos, funciona em conjunto com o emissor IR, que captura os valores de cada ponto, provendo as informações de profundidade do esqueleto (eixo Z). A Figura 4.2 mostra a orientação de rastreamento dos eixos.

Figura 4.2 - Orientação dos eixos do Kinect.



Fonte: KINECT, 2012.

Os quatro microfones funcionando em conjunto são capazes de promover uma sofisticada anulação acústica de eco, supressão de ruído e localização da direção do som. O servo-motor é utilizado para ajustar o ângulo de captura das câmeras, facilitando o enquadramento do corpo (FRANCESE, PASSERO e TORTORA, 2012). Por fim, o sensor Kinect contém um acelerômetro de três eixos configurado para um intervalo de 2g, onde g é a aceleração da gravidade. Isto permite que o sensor relate sua orientação em relação à gravidade e os dados do acelerômetro podem ajudar a detectar quando o sensor é uma orientação incomum (KINECT, 2012). A Tabela 4.1 exibe o conjunto de especificações dos componentes do Kinect.

Tabela 4.1 - Especificação dos componentes do Kinect.

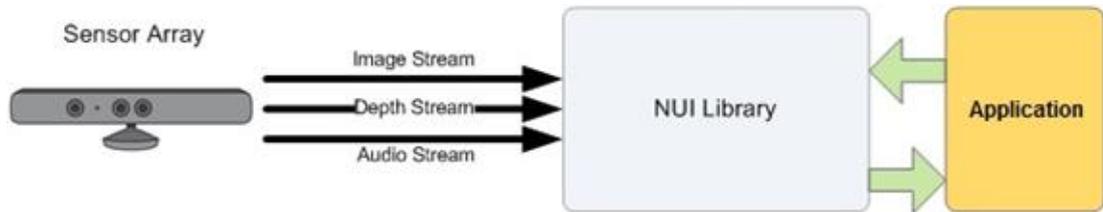
Kinect	Especificação
Ângulo de visão	43° vertical por 57° horizontal
Ângulo de inclinação do servomotor	$\pm 27^\circ$
Taxa de quadros	30 <i>frames per second</i> (FPS)
Formato do audio	16-kHz, 24-bit mono pulse code modulation (PCM)
Características da entrada de audio	Sequência de quatro microfones com conversor analógico-digital (ADC) de 24-bit
Características do acelerômetro	Um acelerômetro, 2g/4g/8g, configurado para 2g com precisão de superior a 1°.

Fonte: KINECT, 2012.

Inicialmente, o Kinect foi projetado para servir como uma interface para jogos do console Xbox, porém, com uma tecnologia tão inovadora, pouco tempo se passou até que uma comunidade de desenvolvedores de software criasse um modo de utilizar o dispositivo através do computador. Em Dezembro de 2010 surgiu o primeiro *Software Development Kit* (SDK) para Kinect em uma versão não oficial (OPENKINECT, 2012).

Em seguida, começaram a surgir softwares que atribuíam ao Kinect as mais diversas funcionalidades, não somente um controle para jogos (VILLAROMAN, ROWE e SWAN, 2011). A Microsoft percebeu o potencial de seu dispositivo como uma interface com o computador e, em Junho de 2011, lançou a versão *beta* oficial, o Kinect for Windows SDK, grátis para usos não comerciais, com suporte técnico e manual de desenvolvimento (KINECT, 2012). Neste trabalho foi utilizada a versão 1.6 do Kinect for Windows SDK, onde é possível manipular todos os dados obtidos pelo Kinect de forma simples e intuitiva. A Figura 4.3 exibe a arquitetura básica da interação de *hardware* e *software* com a aplicação.

Figura 4.3 - Interação de *hardware* e *software* com a aplicação.



Fonte: (KINECT, 2012)

A partir do momento que rastreia o corpo de um indivíduo, o Kinect passa a considerar um esqueleto com 20 pontos-chave, conhecidos como *Joints* (Articulações). Cada articulação possui suas posições nos eixos X, Y e Z. As Figuras 4.4 e 4.5 ilustram o esqueleto rastreado pela câmera RGB e a câmera de profundidade, respectivamente.

Figura 4.4 - Esqueleto rastreado pela câmera RGB.



Fonte: Próprio autor.

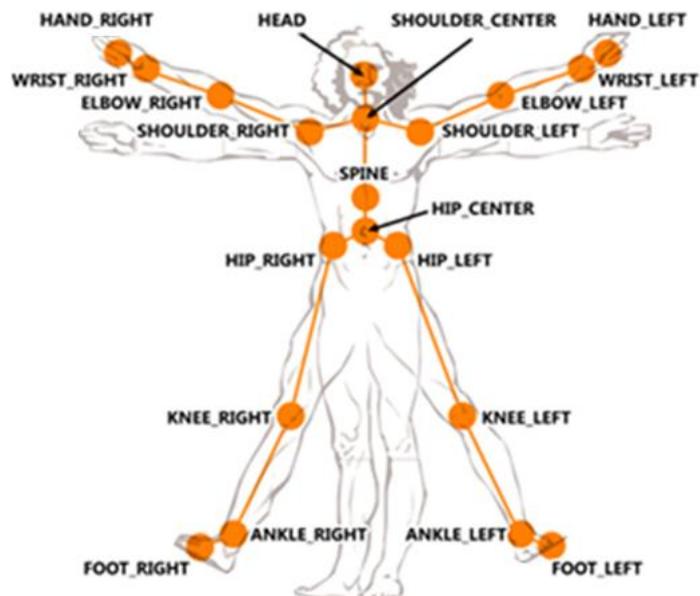
Figura 4.5 - Esqueleto rastreado pela câmera de profundidade.



Fonte: Próprio autor.

Como mencionado, a captura das câmeras do Kinect tem um taxa definida, por padrão, de 30 FPS (*frames per second*). Assim, o desenvolvedor tem à sua disposição um conjunto de 1800 valores por segundo para a manipulação a seu critério e, é através deste manuseio de dados que o reconhecimento de gestos é possível. A Figura 4.6 ilustra todas as articulações detectadas pelo Kinect.

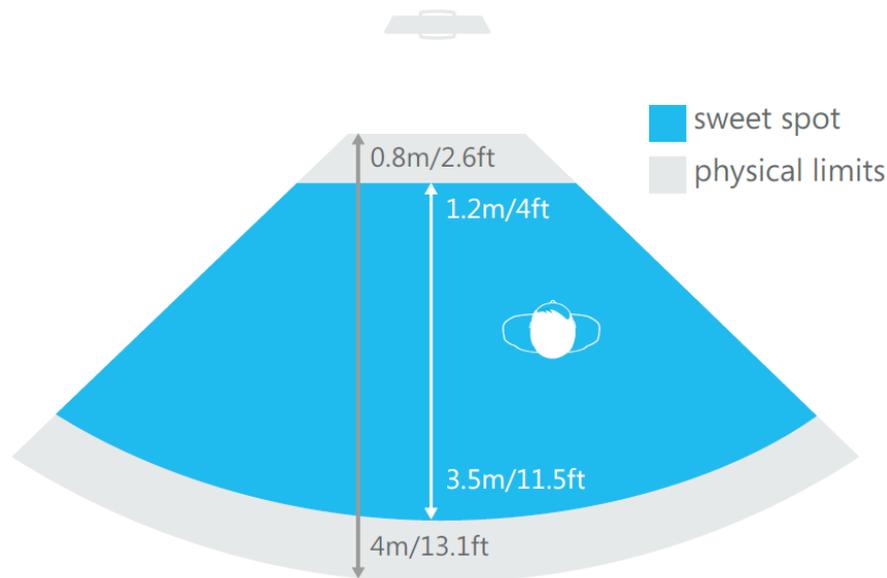
Figura 4.6 - Articulações detectadas pelo Kinect.



Fonte: KINECT, 2012.

Contudo, para que o Kinect consiga rastrear o corpo, é preciso que o mesmo esteja totalmente enquadrado nas câmeras. O intervalo de distância rastreável do Kinect é $[+0.8, +4.0] = \{x \in \mathbb{R} / +0.8 \leq x \leq +4.0\}$ metros, mas o rastreamento é otimizado quando a distância está entre $[+1.2, +3.5] = \{x \in \mathbb{R} / +1.5 \leq x \leq +3.5\}$ metros (KINECT, 2013). A Figura 4.7 mostra os limites físicos e a distância ótima de rastreamento do Kinect.

Figura 4.7 - Distância de rastreamento do Kinect.



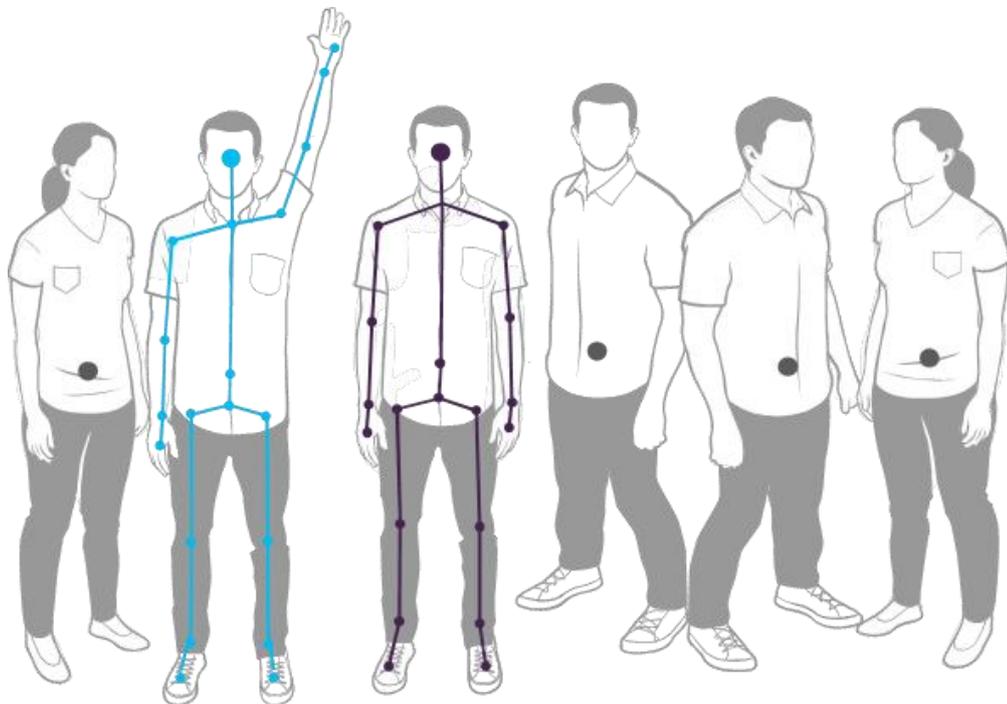
Fonte: KINECT, 2013.

De um lado, existem desvantagens a serem observadas em relação ao Kinect. Devido a falta de marcadores ou sensores junto ao corpo, as informações por vezes podem conter ruídos e imprecisões, cabendo ao desenvolvedor encontrar soluções a estes problemas. Outra desvantagem é que, devido a distância mínima de 1.2 metros para que o Kinect possa rastrear o corpo e trabalhar de forma ideal, pode ser necessário que o usuário possua um projetor de imagens que amplie a tela do computador ou um monitor grande, pois em diversos casos torna-se difícil a leitura e identificação da cena em um monitor pequeno. Além disso, é desconfortável para o desenvolvedor o trabalho de depuração do *software*, uma vez que é necessário levantar-se e posicionar-se a cada execução do código para o perfeito enquadramento do corpo e, caso haja mais pessoas no ambiente, o rastreamento é prejudicado.

Por outro lado, as vantagens do Kinect podem compensar sua utilização, pois a falta de qualquer objeto junto ao corpo é, além de um conforto, às vezes, uma questão de necessidade,

como foi apontado por Gonçalves *et al* (2012). Outra vantagem é a quantidade de informações capturadas: coordenadas X, Y e Z para cada uma das 20 articulações. Desse modo, o reconhecimento de gestos pode ser feito com vários membros do corpo, utilizados individualmente ou combinados. Por último, vale salientar que o Kinect é capaz de rastrear as informações de até dois esqueletos e reconhecer em cena até seis indivíduos ao mesmo tempo, como mostra a Figura 4.8.

Figura 4.8 - Kinect pode reconhecer até seis pessoas e rastrear duas.



Fonte: KINECT, 2013.

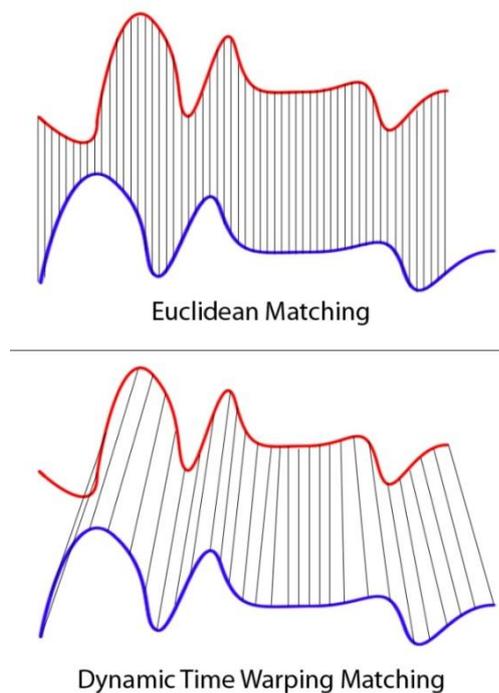
4.2 DYNAMIC TIME WARPING

O *Dynamic Time Warping* (DTW) é bastante conhecido e utilizado em diversas áreas. Este algoritmo surgiu no início dos anos 60 (BELLMAN e KABALA, 1959) e na década de 70 foi extensivamente explorado em aplicações de reconhecimento de voz. Atualmente é utilizado em diversas áreas, como comparação de escritas a mão, assinaturas digitais, reconhecimento de linguagem de sinais, mineração de dados, reconhecimento de gestos, processamentos de sinais, música e tantas outras (SENIN, 2008).

Este algoritmo é baseado em programação dinâmica, medindo a similaridade entre padrões que podem variar o tempo de duração. O conceito principal deste algoritmo é comparar as características de um padrão particular com um modelo de referência previamente estabelecido, ou seja, o DTW compara um padrão de entrada com um padrão de referência e fornece o grau de similaridade entre os dois padrões. Este grau de similaridade, também conhecido como distância, é um resultado numérico e significa que quanto menor o valor obtido, maior a semelhança entre os padrões (GONÇALVES *et al*, 2012).

O DTW estabelece uma correspondência entre duas curvas que estão sujeitas a variações de velocidade, distorcendo o eixo do tempo com compressões e expansões em determinados pontos (NIELS, 2004), conforme mostra a Figura 4.9.

Figura 4.9 - Comparações de curvas entre a distância Euclidiana e DTW.

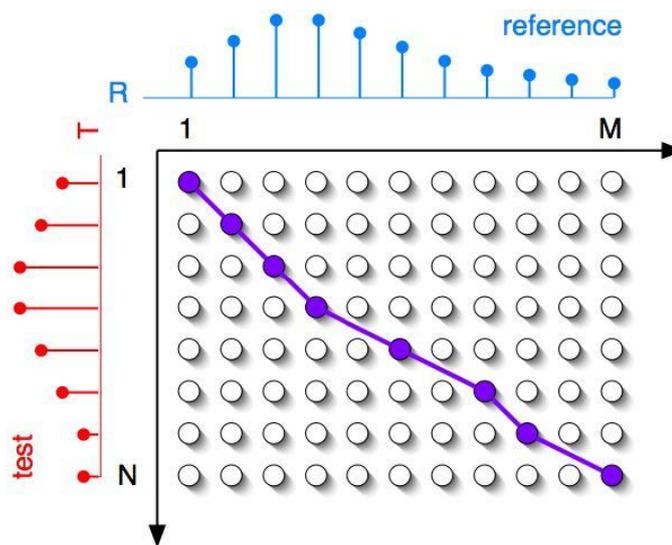


Fonte: WIKIMEDIA, 2013.

Para a semelhança entre dois padrões ou duas curvas serem calculadas, o algoritmo DTW cria uma matriz para armazenar a distância euclidiana de um ponto do padrão de entrada para todos os outros pontos do padrão de referência. Com os valores da matriz resultante, é calculado o melhor caminho, ou seja, o caminho mais curto de uma ponta a outra (KEOGH e PAZZANI, 1999). Para a realização deste cálculo, o primeiro ponto deve necessariamente iniciar em $p_1 = (1, 1)$ e terminar no ponto $p_k = (N, M)$ da matriz, sendo

associado um custo a cada ponto. A distância total entre os vetores comparados será a soma dos custos de cada ponto existente no caminho alcançado pelo algoritmo, sendo que este caminho deverá ser aquele em que a soma dos custos dos pontos seja minimizada (ALMEIDA, 2010). A Figura 4.10 mostra um exemplo de comparação de duas curvas por DTW.

Figura 4.10 - Exemplo de comparação de duas curvas por DTW.



Fonte: JEHAN, 2005.

Matematicamente, o caminho de alinhamento construído pelo DTW é uma sequência de pontos $p = (p_1, p_2, \dots, p_k)$ com $p_l = (p_i, p_j) \in [1 : N] \times [1 : M]$ para $l \in [1 : K]$ que deve satisfazer os seguintes critérios (SENIN, 2008):

- Condição de contorno: $p_1 = (1, 1)$ e $p_k = (N, M)$. Ou seja, o início e o fim do caminho devem ser o primeiro e o último ponto da sequência, respectivamente.
- Condição de monotonicidade: $n_1 \leq n_2 \leq \dots \leq n_k$ e $m_1 \leq m_2 \leq \dots \leq m_k$. Essa condição preserva a ordem dos pontos em relação ao tempo, ou seja, o caminho não pode possuir um trajeto em que volte no tempo.
- Condição de continuidade: $p_{l+1} - p_l \in \{(1, 1), (1, 0), (0, 1)\}$. Este critério exige que o caminho gerado seja contínuo, ou seja, um ponto está necessariamente ligado a um ponto adjacente na sequência.

5 PROJETO E IMPLEMENTAÇÃO

Neste capítulo serão mostrados em detalhes o projeto e implementação do Kinect no Sistema ITV para o reconhecimento de gestos, controle da interface e navegação no ambiente.

5.1 RECONHECIMENTO DE GESTOS

A escolha do *Dynamic Time Warping* para o reconhecimento de gestos deve-se aos resultados satisfatórios obtidos por Gonçalves *et al* (2012) e Doliotis *et al* (2011). Estes estudos utilizam a mesma metodologia: Kinect com DTW, onde as taxas de reconhecimento se aproximam, em média, dos 90% de acerto. Além disso, a título de comparação com os resultados obtidos por Fonseca (2013) com o *Wii Remote Controller*, foi utilizada a mesma técnica e dois movimentos semelhantes.

Os três gestos a serem reconhecidos são os movimentos de chave de boca, chave de fenda e martelo. O movimento de chave de boca consiste em estender a mão frente ao corpo e recolhê-la junto ao corpo, em uma leve curvatura aberta em relação ao tronco. A movimentação da chave de fenda é feita estendendo a mão frente ao corpo e rotacionando-a no sentido horário. Por fim, o gesto que representa a movimentação de martelo consiste em, com o cotovelo imóvel, erguer e baixar a mão até a altura do ombro por duas vezes. Todos os gestos devem ser executados com a mão direita.

5.1.1 Normalização

O sensor Kinect trabalha com valores reais de $[-1.0, +1.0] = \{x \in \mathbb{R} / -1.0 \leq x \leq +1.0\}$ nos eixos X e Y, indicando os limites que a câmera consegue capturar e de $[+0.8, +4.0] = \{x \in \mathbb{R} / +0.8 \leq x \leq +4.0\}$ no eixo Z, indicando a distância real em metros.

Se, por exemplo, um gesto tido como modelo de referência, feito na região positiva de X e Y, for comparado com um movimento idêntico, feito na região negativa de X e Y, o resultado será um reconhecimento inválido, pois para o algoritmo DTW, os valores por mais que sigam o mesmo padrão, são muito diferentes.

Portanto, foi criado um método que normaliza todos os padrões coletados pelo Kinect para que o usuário possa executar seu movimento em qualquer lugar do ambiente, desde que seu corpo esteja sendo rastreado. Este método faz a diferenciação sequencial entre os elementos de um vetor unidimensional e retorna o vetor resultante com um elemento a menos. As Equações 5.1 e 5.2 ilustram um exemplo de normalização de um vetor.

Equação 5.1 - Vetor original.

$$[x_1, x_2, x_3, x_4, x_5, x_6] = [2, 3, 5, 9, 5, 5]$$

Fonte: Próprio autor.

Equação 5.2 - Vetor normalizado.

$$[x_2 - x_1, x_3 - x_2, x_4 - x_3, x_5 - x_4, x_6 - x_5] = [1, 2, 4, -4, 0]$$

Fonte: Próprio autor.

5.1.2 Detecção de Movimentos

O Kinect é capaz de rastrear 20 articulações, mas para os gestos utilizados neste trabalho, como os movimentos de martelo, chave de boca e chave de fenda, poucas articulações do corpo se movimentam. Desta forma, as articulações que menos se movimentam, como as pernas, tronco, cabeça e mão esquerda foram descartadas e, assim, é possível trabalhar com uma menor quantidade de dados para que o reconhecimento de gesto seja mais rápido.

Foram definidas para o rastreamento apenas 3 articulações: mão, pulso e cotovelo. Com essas 3 *Joints* é possível o reconhecimento de gesto com poucos erros e a quantidade de dados a manipular é menor, aumentando a rapidez da resposta do sistema.

Em seguida, foi desenvolvido um *software* auxiliar para coletar 15 amostras de gestos, com tempo fixo de 2 segundos cada, e salvá-las em arquivos. Dessas amostras, uma foi utilizada como padrão de referência e as outras 14 como conjunto de teste submetido ao algoritmo DTW. Com a média desses resultados, espera-se obter o valor ideal do limiar de reconhecimento. A Tabela 5.1 mostra os resultados obtidos pelo algoritmo DTW para o movimento de martelo na articulação da mão.

Tabela 5.1 - Saída do DTW para movimento de martelo na articulação da mão.

Amostra	Eixo X	Eixo Y	Eixo Z
1	0.279	0.829	0.748
2	0.347	0.294	0.577
3	0.311	0.384	0.745
4	0.760	1.573	0.921
5	0.324	0.637	0.751
6	0.465	0.533	0.882
7	0.349	0.686	0.978
8	0.209	0.643	0.440
9	0.420	0.875	0.601
10	0.102	0.534	0.394
11	0.419	0.992	0.655
12	0.916	3.202	1.829
13	0.283	1.003	0.870
14	0.381	0.674	0.993

Fonte: Próprio autor.

De posse desses dados, é possível encontrar um limiar de reconhecimento de gestos para o DTW, ou seja, se o valor do padrão de entrada for menor que o limiar estabelecido, este é reconhecido como gesto válido, caso contrário, é considerado inválido.

Foram consideradas duas abordagens para a definição do limiar. A primeira consistiu em definir o limiar de cada eixo como o maior valor obtido entre as amostras. A segunda abordagem baseou-se em utilizar a média de todas as amostras.

Com a primeira abordagem, o reconhecimento do gesto se tornaria pouco confiável, pois movimentos não tão semelhantes, ainda assim, seriam considerados certos. Isso acontece porque algumas amostras podem conter ruídos e dados incorretos, gerando uma saída numérica alta pelo algoritmo DTW.

A segunda abordagem restringe mais o reconhecimento do gesto, desta forma, a amostra deve ser realmente parecida com o padrão de referência para que seja considerada correta. Portanto, a média das amostras foi definida como os limiares de cada eixo. Calculando os valores, foram encontradas para os eixos X, Y e Z, as médias de 0.398, 0.918 e 0.813, respectivamente. O mesmo processo foi estendido para as outras articulações e para os

outros gestos, a fim de se obter todos os limiares ideais. A Tabela 5.2 exibe os limiares encontrados de todas as articulações em seus respectivos eixos.

Tabela 5.2 - Limiares definidos.

Movimento	Articulação	Eixo X	Eixo Y	Eixo Z
Martelo	Mão	0.398	0.918	0.813
	Pulso	0.412	0.877	0.628
	Cotovelo	0.357	0.482	0.539
Chave de Fenda	Mão	0.443	0.298	0.419
	Pulso	0.449	0.303	0.530
	Cotovelo	0.576	0.514	0.583
Chave de Boca	Mão	0.510	0.289	0.742
	Pulso	0.662	0.355	0.783
	Cotovelo	0.458	0.179	0.674

Fonte: Próprio autor.

Para o reconhecimento completo de um gesto é necessário que o algoritmo DTW seja executado 9 vezes, ou seja, 3 vezes para cada eixo de cada uma das 3 articulações. Portanto, mesmo que o DTW seja método direto, é perceptível um pequeno atraso na resposta. O tempo definido para a realização de cada gesto é de 2 segundos e o Kinect fornece informações 30 vezes por segundo. Sendo assim, são 60 dados em cada eixo de cada articulação, totalizando 540 dados a serem manipulados por amostra.

Visando diminuir o atraso no tempo de reconhecimento, foi realizado um ajuste na quantidade de dados adquiridos, alterando o número de valores em cada padrão. Neste ajuste, manteve-se o aspecto correto da curva, ignorando alguns elementos da sequência, de forma intercalada e periódica, por exemplo: na sequência de 1 a 10, são selecionados apenas os elementos pares. A Tabela 5.3 mostra os resultados dos testes com novas 20 amostras para o movimento de martelo da mão.

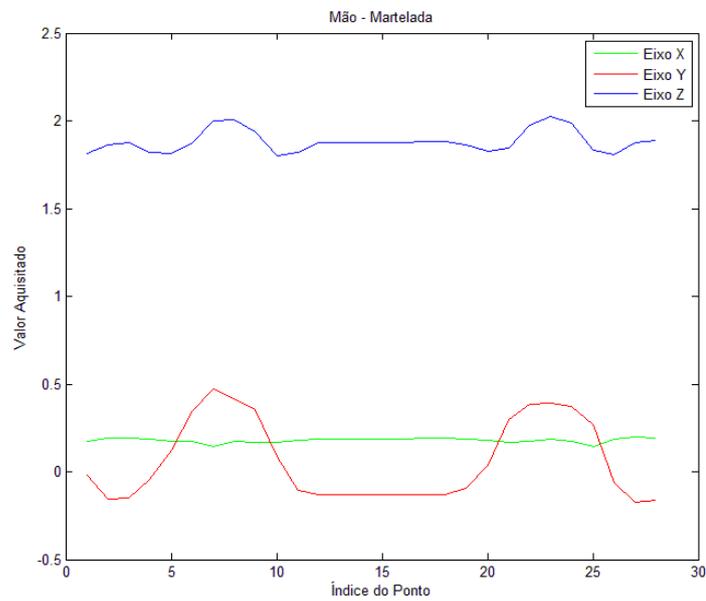
Tabela 5.3 - Teste da quantidade de valores por padrão.

Quantidade de valores por padrão	Acertos (%)
60	85%
30	85%
20	80%
10	60%

Fonte: Próprio autor.

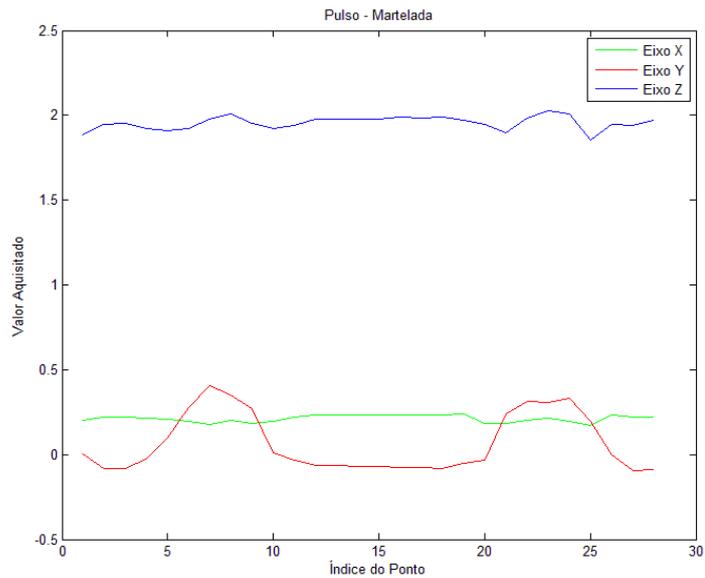
Por reduzir pela metade a quantidade de valores por padrão e, mesmo assim, continuar com o maior índice de acertos, foi definida a quantidade de 30 valores por amostra. Com esta alteração, o conjunto de dados utilizados por amostra se tornou 270 e o atraso na resposta de reconhecimento do gesto foi inferior. As Figuras 5.1, 5.2 e 5.3 mostram gráficos da movimentação de martelo nos três eixos de cada articulação.

Figura 5.1 - Gráfico dos dados da mão



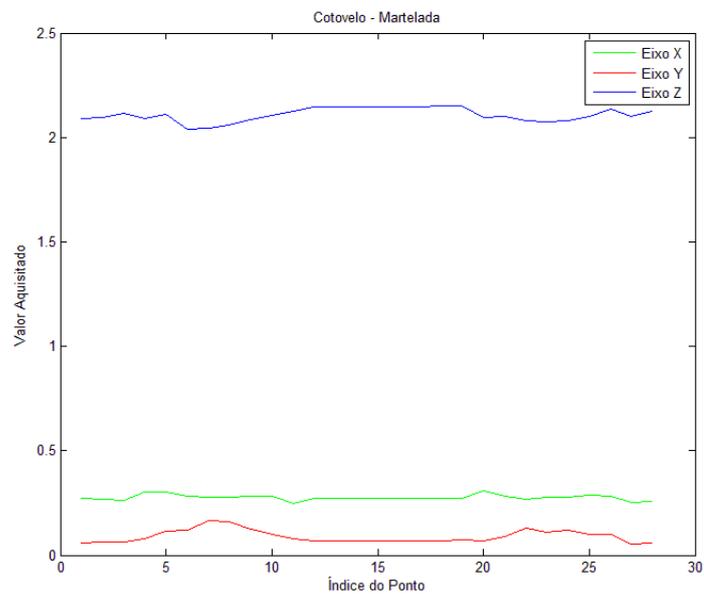
Fonte: Próprio autor.

Figura 5.2 - Gráfico dos dados do pulso



Fonte: Próprio autor.

Figura 5.3 - Gráfico dos dados do cotovelo.



Fonte: Próprio autor.

Nas figuras acima, pode-se observar características interessantes quanto ao rastreamento do movimento, por exemplo: as curvas em vermelho indicando o eixo Y (movimento vertical) variam bastante na articulação da mão, atenuam um pouco no pulso e variam pouco no cotovelo. Estas características fazem sentido se for observado o movimento

de uma martelada, pois a mão sobe e desce em um movimento longo, o pulso em um movimento menor e o cotovelo permanece praticamente estático.

Além disso, pode-se perceber que a linha verde, indicando o eixo X, ou seja, a posição horizontal, pouco varia em todas as articulações, pois o movimento de martelo é basicamente vertical. Assim, existe a possibilidade de descartar o eixo X, aumentando a taxa de reconhecimento do gesto com o limiar definido anteriormente, porém, isso pode acarretar em falso-positivos. Movimentos de martelo que contém variações horizontais são diferentes do padrão de referência e não devem ser considerados como válidos. Desta forma, por mais que o eixo X varie pouco, é importante para o reconhecimento preciso do gesto, assim como os outros dois eixos.

Vale lembrar que, pelo fato do Kinect reconhecer somente a posição das articulações, e não a rotação delas, é possível que em alguns casos o reconhecimento de gestos retorne um falso-positivo. Para elucidar o fato, toma-se o exemplo da movimentação de chave de fenda, onde mesmo que a mão rotacione no sentido horário, anti-horário ou até mesmo permaneça parada, para o Kinect, a articulação não sofreu considerável variação de posição. Esta é uma das vantagens do *Wii Remote Controller*, implementado no Sistema ITV por Fonseca (2013), que conta com acelerômetros de rotação nos três eixos.

5.1.3 Hierarquia de Reconhecimento

Como mencionado, para o reconhecimento completo de um gesto é necessário que todos os padrões das articulações em todos os eixos retornem, pelo algoritmo do DTW, um número menor que os limiares estabelecidos. Tendo em vista um reconhecimento mais preciso e rápido, foi determinada uma hierarquia de articulações, ou seja, uma articulação depende da outra para ser verificada.

Foram testadas e verificadas, com 15 novas amostras, duas possibilidades de hierarquia com as 3 articulações trabalhadas. A primeira possibilidade inicia-se com a mão, em seguida o pulso e por fim o cotovelo. A segunda é o caminho inverso. A Tabela 5.4 mostra os resultados encontrados.

Tabela 5.4 - Resultado do teste de hierarquia.

Hierarquia	Acerto (%)
Mão > Pulso > Cotovelo	86.67
Cotovelo > Pulso > Mão	86.67

Fonte: Próprio autor.

Esse resultado mostra que não houve diferença na precisão do reconhecimento dos gestos, o que faz sentido, pois se uma amostra é dada como válida, não importa a sequência em que ela é comparada. Contudo, é importante utilizar uma das duas hierarquias, pois evita cálculos desnecessários, fornecendo uma resposta mais rápida ao usuário. Por exemplo, se na etapa de comparação de padrões do pulso, o limiar definido não for atingido, as etapas seguintes são ignoradas e o gesto é considerado inválido. A hierarquia definida foi iniciada pela articulação da mão e finalizada com o cotovelo.

Por fim, foram criadas 15 amostras para o movimento de chave de fenda e 15 para o movimento de chave de boca, a fim de validar o reconhecimento dos gestos. A Tabela 5.5 mostra o resultado da validação.

Tabela 5.5 - Resultado da validação.

Movimento	Acertos (%)
Martelo	86.67
Chave de Fenda	86.67
Chave de Boca	73.33

Fonte: Próprio autor.

5.2 CONTROLE DA INTERFACE

Uma das desvantagens do Kinect é que, pelo seu conceito, não existem botões ou teclas a serem acionadas. Desta forma, para que haja uma melhor interação entre o usuário e o sistema e, para que não haja a necessidade de se locomover entre a área de reconhecimento de gestos e a área onde estão o mouse e o teclado, foi desenvolvida uma forma de controle da interface do Sistema ITV pela simulação do mouse (cursor e botão) através do sensor Kinect.

Para a implementação destas funções, foi utilizada uma API nativa do Windows para C++, denominada `windows.h`. Esta biblioteca possui várias funções para controle de interfaces com o usuário, como mouse e teclado.

5.2.1 Cursor do Mouse

Entre os métodos da API, existe uma função que posiciona o mouse em um determinado local da tela através de dois argumentos, passados como parâmetros, que representam os valores dos *pixels*, horizontal e vertical, respectivamente, onde o cursor do mouse deve se situar. Esta função foi sincronizada com o rastreamento da posição horizontal e vertical da mão direita.

Porém, para que seja possível acessar todos os *pixels* da tela, é preciso que seja feita uma normalização dos valores obtidos. O Kinect retorna os valores dos eixos X e Y em um intervalo de $[-1.0, +1.0] = \{x \in \mathbb{R} / -1.0 \leq x \leq +1.0\}$. Desta forma, o primeiro passo na normalização é somar 1 unidade a cada valor coletado, transformando-o em um valor positivo. Em seguida, para facilitar a compreensão, este mesmo valor é dividido por 2, deste modo, o número resultante está contido no intervalo de $[0.0, +1.0] = \{x \in \mathbb{R} / 0.0 \leq x \leq +1.0\}$.

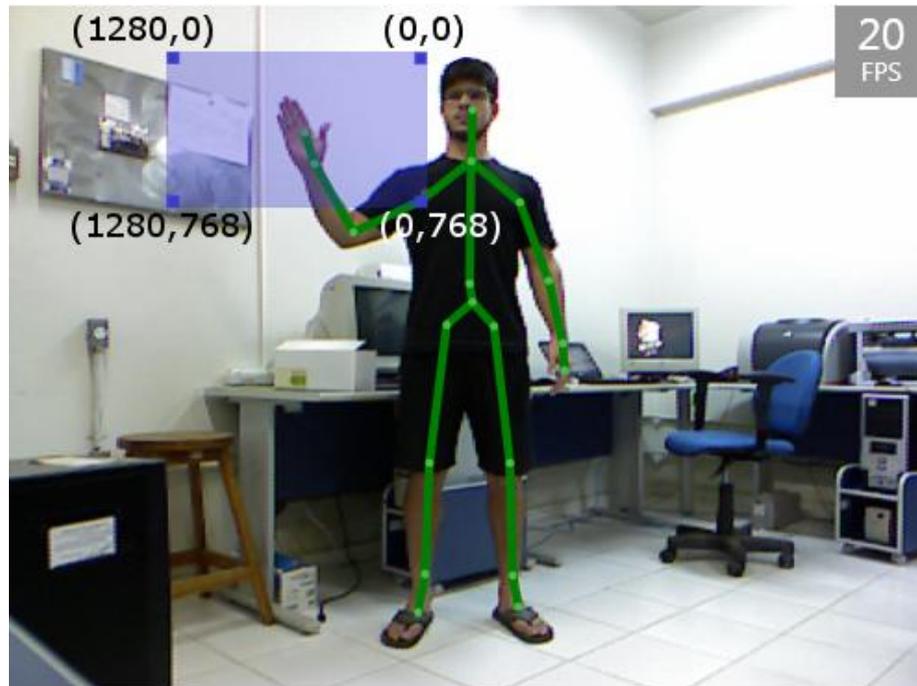
Utilizando uma função de detecção da resolução da tela, é possível configurar que os valores obtidos pelo Kinect representem os limites da tela. Para isso, basta passar para a função controladora do mouse a multiplicação do valor adquirido pela resolução da tela em seu respectivo eixo. Terminado este ajuste, o cursor do mouse pode ser posicionado por toda a tela através do rastreamento da mão direita.

Ainda assim, há uma dificuldade a ser enfrentada, pois para que o usuário atinja os quatro cantos da tela, é necessário que a posição da mão alcance os limites da câmera do Kinect, o que pode significar um desafio. O campo de visão da câmera pode ser grande, a depender da distância do indivíduo em relação ao dispositivo. Deste modo, foi definida uma região de rastreamento, proporcional à resolução de tela detectada. Esta área é ajustada restringindo-se o espaço de rastreamento que manipula o mouse e, em seguida, são feitos cálculos para que os limites da região definida sejam equivalentes aos da resolução da tela.

A área deliberada deve situar-se ao lado direito do corpo, entre o tronco e a cabeça. A Figura 5.4 ilustra o rastreamento da mão direita, do ponto de vista do Kinect, para o

posicionamento do cursor do mouse referente ao número de *pixels* em um monitor com resolução de 1280x768 px.

Figura 5.4 - Visão da câmera no rastreamento da posição do mouse.



Fonte: Próprio autor.

Como mencionado, o Kinect é capaz de fornecer informações a uma taxa de 30 vezes por segundo, ou seja, 30 Hertz (Hz). Com essa frequência de ajuste de posição do mouse e sabendo que as informações do Kinect podem ser, às vezes, ruidosas, o cursor do mouse acaba não estabilizando em uma posição, ou seja, o ponteiro do mouse constantemente treme. Isso pode dificultar consideravelmente a seleção de objetos da cena.

Deste modo, a solução encontrada foi a adição de um filtro passa-baixas no código de movimentação do mouse, para que o mesmo permanesse menos sensível aos ruídos dos dados do Kinect. O filtro passa-baixas permite a passagem de baixas frequências sem dificuldades e atenua a amplitude das frequências maiores que a frequência de corte. A quantidade de atenuação para cada frequência varia de filtro para filtro (MOTA, 2011).

Para suprimir ruídos e reduzir os erros, este filtro passa-baixas também foi utilizado nos valores coletados para os padrões de referência e todos os padrões de entrada do reconhecimento de gestos descrito na seção 5.1.

O filtro utilizado calcula uma média ponderada em relação ao elemento anterior da sequência, evitando assim, uma alta variação entre dois elementos em caso de ruído. A Equação 5.3 define o filtro implementado.

Equação 5.3 - Filtro passa-baixas.

$$x_i = x_i \cdot (1 - \alpha) + x_{i-1} \cdot \alpha, \quad \alpha \in \mathbb{R} / 0.0 \leq \alpha \leq +1.0$$

Fonte: MOTA, 2011.

Na Equação 5.3, x_i é o resultado a ser utilizado como a posição atual do mouse, x_{i-1} é o valor da posição anterior e o α (alfa) é chamado de parâmetro ou coeficiente do filtro. Este coeficiente foi empiricamente ajustado para +0.85 após vários testes. Desta forma, a variação de posição é atenuada e o ponteiro estabilizado. A Figura 5.5 exibe o algoritmo utilizado para o controle do mouse.

Figura 5.5 - Algoritmo de controle do mouse.

```
//Normalização dos dados obtidos pelo Kinect para o intervalo [0.0, +1.0]
a1 = (1+skel.SkeletonPositions[NUI_SKELETON_POSITION_HAND_RIGHT].x)/2;
a2 = (1+skel.SkeletonPositions[NUI_SKELETON_POSITION_HAND_RIGHT].y)/2;

//Condicional que restringe a área reconhecida proporcionalmente à resolução da tela
if(a1 >= 0.5 & a1<= ( 0.5 + (Width*0.00015) ) & a2 >= 0.7 & a2 <= ( 0.7 + (Heigth*0.00015) ) ){

    //Normalização da área restrita para o intervalo [0.0, +1.0]
    b1[1] = (a1 - 0.5) / (Width*0.00015);
    b2[1] = (a2 - 0.7) / (Heigth*0.00015);

    //Obtenção dos pixels referentes na tela
    b1[1] = b1[1]*Width;
    b2[1] = b2[1]*Heigth;

    //Módulo do eixo Y
    b2[1] = fabs(b2[1]-Heigth);

    //Aplicação do filtro passa-baixas com alfa = 0.85
    b1[1] = b1[1]*(1-alfa) + b1[0]*alfa;
    b2[1] = b2[1]*(1-alfa) + b2[0]*alfa;

    //Método que posiciona o mouse
    SetCursorPos((int)b1[1],(int)b2[1]);

    //Guarda o valor para o próximo uso.
    b1[0] = b1[1];
    b2[0] = b2[1];
}
```

Fonte: Próprio autor.

5.2.2 Clique do Mouse

Após o posicionamento do mouse através do Kinect ter sido devidamente ajustado, foi desenvolvida a função de clique do botão esquerdo do mouse. O botão esquerdo geralmente é utilizado para selecionar o objeto de interação com o usuário no Sistema ITV. Sendo assim, é de grande importância sua implementação. Novamente, utilizando uma das funções da biblioteca windows.h, é possível simular o clique do mouse facilmente.

Para que o clique seja executado, foi definido um gesto com a mão esquerda, onde o usuário deve erguer a mão até a altura do ombro e descê-la novamente. Esta função é implementada através do rastreamento da posição da mão esquerda, que aciona o clique se o valor da posição no eixo Y for superior a um determinado valor. Este valor foi definido empiricamente em +0.3, ou seja, se o rastreamento da mão esquerda retornar um valor maior que +0.3, a função de clique será acionada.

Todavia, como a frequência de informações do Kinect é de 30 Hz, entre o tempo da mão do usuário subir até o ombro e descer novamente, podem acontecer vários cliques desnecessários, atrapalhando a execução do sistema. Portanto, foi criado um contador de rastreamentos para aumentar o tempo entre um clique e outro, ou seja, a função de clique é ativada apenas depois de certa quantidade de valores coletados acima de +0.3. Este contador foi definido com o valor de 10 unidades, ou seja, de acordo com a frequência do Kinect, 0.333 segundo de intervalo entre um clique e outro. A Figura 5.6 exibe o algoritmo utilizado na função de clique.

Figura 5.6 - Algoritmo utilizado na função de clique do mouse.

```
// Condicional de posição da mão esquerda
if (skel.SkeletonPositions[NUI_SKELETON_POSITION_HAND_LEFT].y >= 0.3){

    //Incremento do contador
    contador_mouse++;

    //Condicional de verificação do contador
    if (contador_mouse == 10){

        //Função que implementa o clique do mouse
        mouse_event(MOUSEEVENTF_LEFTDOWN | MOUSEEVENTF_LEFTUP, 0, 0, 0, 0);

        //Reinício do contador
        contador_mouse = 0;
    }
}
```

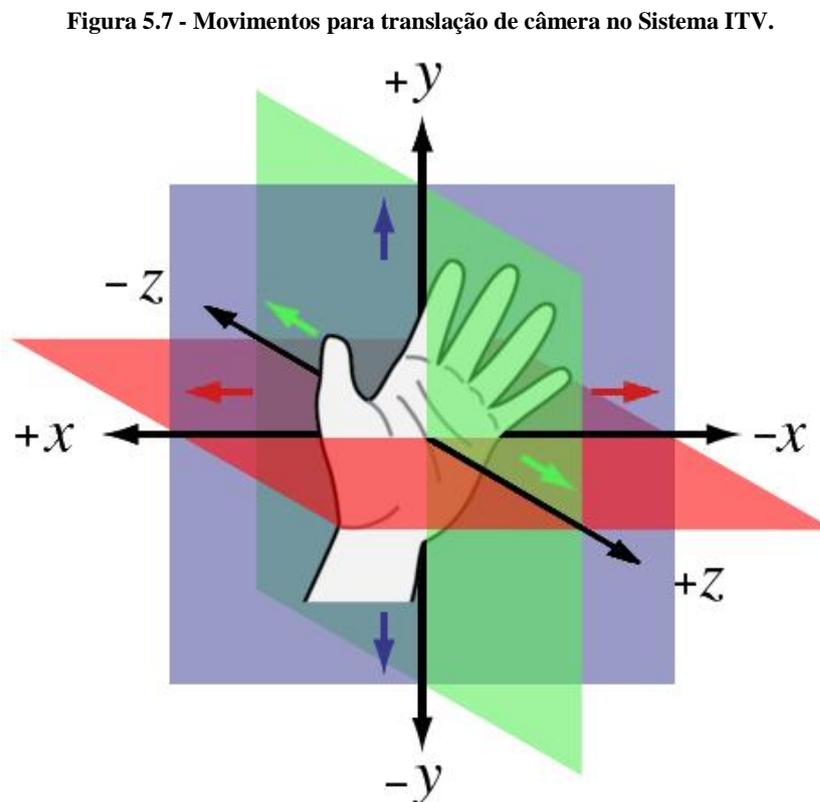
Fonte: Próprio autor.

5.3 NAVEGAÇÃO NO AMBIENTE

Na execução de uma ITV, há a possibilidade de, a qualquer momento, pausar a animação e navegar livremente pelo ambiente virtual através de mouse e teclado. Essa navegação consiste, basicamente, na translação e rotação da câmera de cena, renderizada pelo motor gráfico Ogre3D. Isso permite que a visualização de ambientes, ferramentas e equipamentos, seja melhor.

Foi desenvolvida uma navegação pelo ambiente por meio do rastreamento da posição das mãos pelo sensor Kinect. Definiu-se que a mão esquerda do usuário controla a posição da câmera nas três dimensões, substituindo o teclado e, a mão direita controla a rotação da câmera, ou seja, para qual direção a câmera deve apontar, substituindo o mouse.

Foram determinadas regiões de rastreamento (do mesmo modo que a Figura 5.4, seção 5.2.1, deste capítulo) para ambas as mãos. A Figura 5.7 ilustra, do ponto de vista do Kinect, os possíveis movimentos da mão esquerda para a translação da câmera, em coordenadas referentes ao sistema de eixos do Ogre3D.

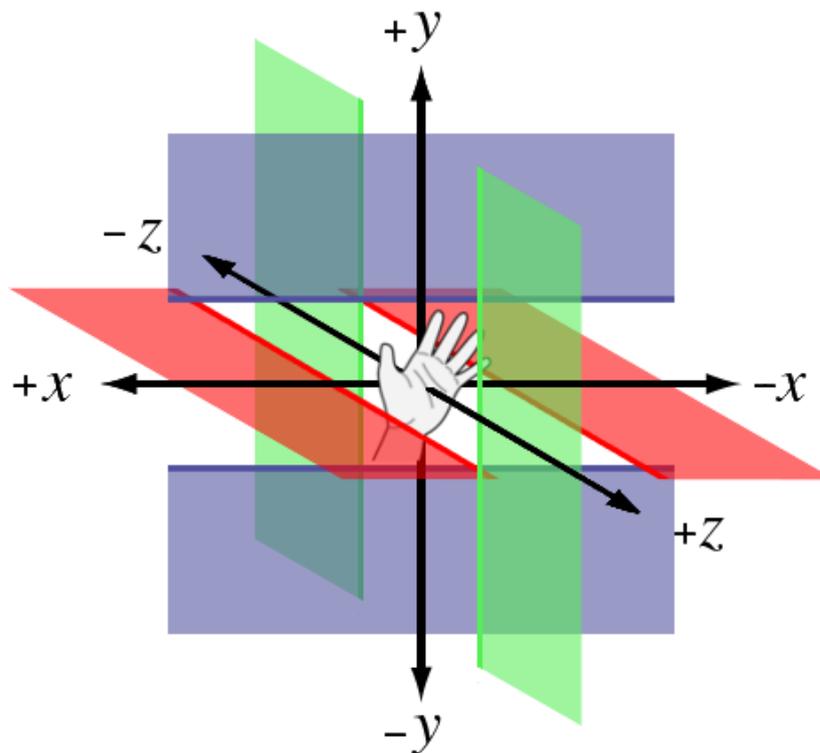


Fonte: Próprio autor.

Portanto, se a mão esquerda se movimentar pela região em verde (eixo Z), ação da câmera será de aproximar-se ou afastar-se da cena, caso percorra a região em vermelho (eixo X), a câmera fará a translação para os lados e, caso a mão esquerda se movimente na região em azul (eixo Y), o movimento da câmera será para cima ou para baixo.

Cada região foi segmentada em três partes iguais para que haja a possibilidade da câmera não se movimentar, ou seja, uma região neutra. Assim, se o usuário desejar parar a cena da navegação, ele deve posicionar a mão no centro dessas regiões para que a câmera fique estática. A Figura 5.8 exemplifica essa situação.

Figura 5.8 - Região neutra de posicionamento da câmera.

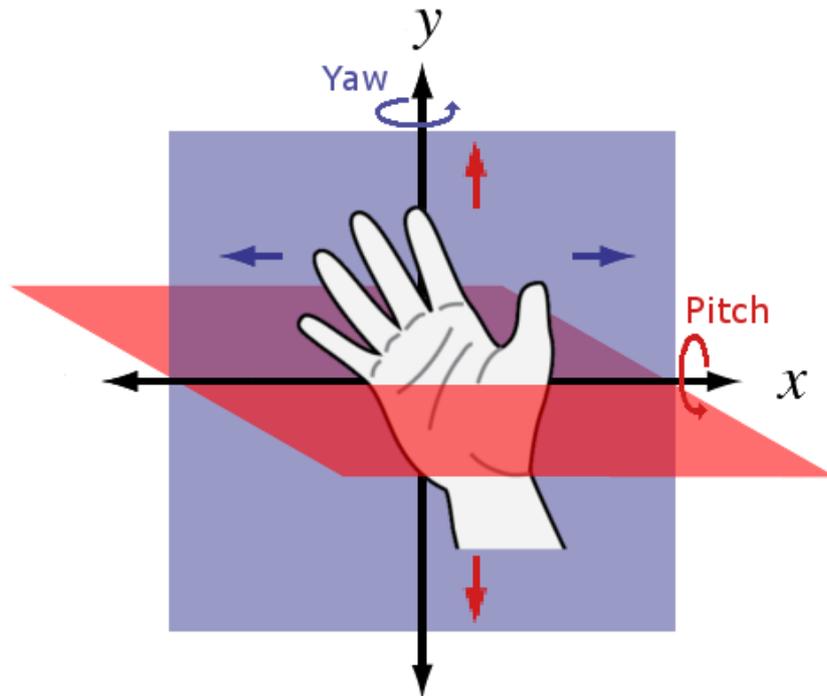


Fonte: Próprio autor.

Vale ressaltar que, no eixo Z (região verde), a posição obtida pela mão é em relação a articulação do quadril, desta forma, o usuário pode se movimentar livremente para frente ou para trás e, mesmo assim, a distância que sua mão precisa ser levada à frente do seu corpo permanece igual. O mesmo não acontece com os eixos X e Y, ou seja, eles não tem como referência o quadril, e sim uma região fixa. Esta alternativa não foi adotada para os eixos X e Y devido à soma de ruídos nas articulações que podem provocar uma navegação mais difícil, porém, para o eixo Z, esse artifício é indispensável.

A mão direita é a responsável pela rotação da câmera. Como no Sistema ITV não há rotação da câmera em torno do eixo Z (*Roll*), são necessários apenas dois eixos de manipulação de rotação: X e Y . Deste modo, foram definidas regiões de rastreamento para a mão direita nesses dois eixos. A Figura 5.9 exhibe as possibilidades de movimentos da mão direita para a rotação da câmera em coordenadas do Ogre3D.

Figura 5.9 - Movimentos para a rotação de câmera no Sistema ITV.



Fonte: Próprio autor.

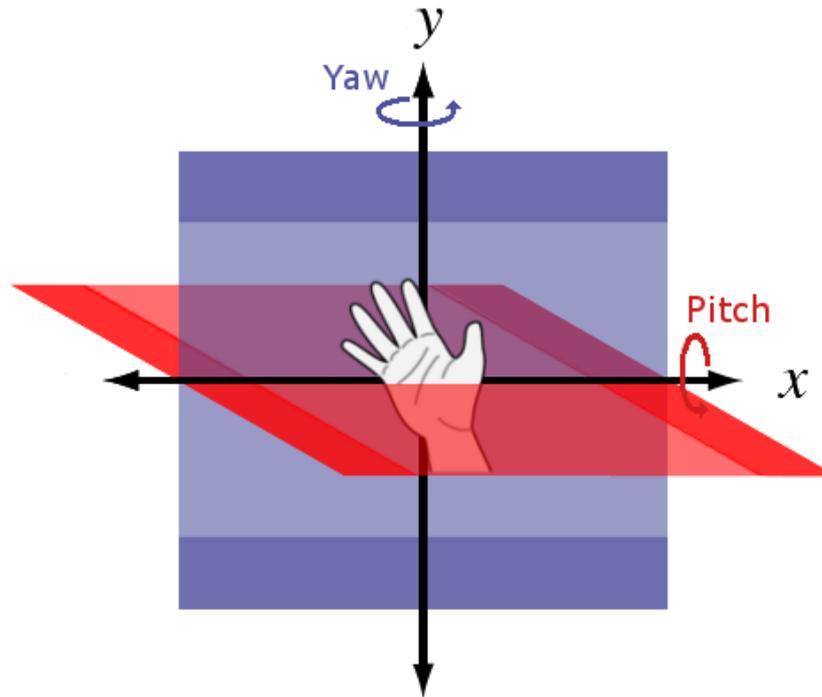
Pode-se observar que a rotação da câmera em torno do eixo Y (*Yaw*) é feita com movimentos laterais, ou seja, deslocamentos na região em vermelho (eixo X). A rotação da câmera em torno do eixo X (*Pitch*) é feita através de movimentos para cima e para baixo, ou seja, variações na região azul (eixo Y).

Com o objetivo de substituir o funcionamento do mouse e tornar a navegação mais atraente, não foi definida uma região neutra para a mão direita, já que, no Sistema ITV, ao navegar pelo mouse, a câmera é rotacionada continuamente por seus movimentos.

Porém, essa escolha pode impedir que o usuário consiga rotacionar a câmera em um ângulo de 360° , pois há a possibilidade da movimentação dentro da área restrita de rastreamento da mão não ser o suficiente. Por este motivo, cada uma das regiões de rotação

foi segmentada em 3 partes, sendo: 70% a região central, onde a rotação simula a navegação pelo mouse e 15% de cada lado, onde a rotação permanece constante até que a mão retorne à região central. Desta forma, é possível que o usuário realize voltas completas pelo ambiente virtual. A Figura 5.10 ilustra as regiões segmentadas em seus respectivos eixos de rotação.

Figura 5.10 – Regiões de segmentadas para rotação da câmera.



Fonte: Próprio autor.

As regiões mais claras, ou seja, centrais, representam a rotação da câmera simulando o mouse. As áreas escuras, ou seja, periféricas, ilustram a rotação em velocidade constante da câmera.

Deste modo, o usuário pode navegar no ambiente virtual do Sistema ITV através da movimentação de suas duas mãos. A Figura 5.11 ilustra, do posto de vista do Kinect, o usuário e as possibilidades de movimentação da câmera.

Figura 5.11 - Possibilidades de movimentação da câmera.



Fonte: Próprio autor.

Como mencionado, o controle da câmera é feito através da biblioteca Ogre3D, responsável pela renderização da cena no Sistema ITV. Assim, tanto mouse e teclado, quanto o dispositivo Kinect utilizam funções do Ogre3D para esse controle. A Tabela 5.6 mostra a equivalência entre as funções e as interfaces de entrada.

Tabela 5.6 - Equivalência entre funções e interfaces de entrada.

Câmera Ogre3D	Mouse	Teclado	Kinect
Rotação eixo Y (Yaw)	Movimentos esquerda/direita	-	Movimentos horizontais da mão direita
Rotação eixo X (Pitch)	Movimentos frente/trás	-	Movimentos verticais da mão direita
Rotação eixo Z (Roll)	-	-	-
Translação eixo X	-	Setas direcionais esquerda/direita	Movimentos horizontais da mão esquerda
Translação eixo Y	-	Teclas <i>Page Up</i> e <i>Page Down</i>	Movimentos verticais da mão esquerda
Translação eixo Z	Botão direito/meio	Setas direcionais cima/baixo	Movimentos frente/trás da mão esquerda

Fonte: Próprio autor.

Para a ativação do modo de navegação pelo Kinect, é preciso que o usuário erga os dois braços à altura do ombro durante 1.5 segundos. Feito isso, inicia-se o modo de navegação e o usuário pode percorrer livremente o ambiente virtual. Para sair do modo de navegação e voltar ao modo de controle de interface, basta que o indivíduo baixe um dos braços.

5.4 INTEGRAÇÃO DO KINECT AO SISTEMA ITV

5.4.1 Classes e Bibliotecas

Com o reconhecimento de gestos, controle da interface e navegação estruturados, iniciou-se o processo de integração das informações coletadas pelo Kinect ao Sistema ITV. Porém, uma enorme dificuldade foi encontrada: a falta de documentação do *software* Sistema ITV.

A documentação de um *software* é uma subárea da engenharia de *software* que, segundo Rezende (2005), é uma metodologia de desenvolvimento e manutenção de sistemas modulares com várias características, entre elas:

- Processo (roteiro) dinâmico, integrado e inteligente de soluções tecnológicas;
- Adequação aos requisitos funcionais do cliente e seus respectivos procedimentos pertinentes;
- Planejamento e gestão de atividades, recursos, custos e datas;
- Efetivação de padrões de qualidade, produtividade e efetividade em suas atividades e produtos.

Esta documentação é de fundamental importância para propósitos de futuras manutenções e aprimoramentos, sendo indispensável para uma rápida compreensão da estrutura de um aplicativo de médio e grande porte. Em sua essência, deve conter, entre outros: documentos com especificações, diagramas de classe e fluxo de dados (PRESSMAN, 2006). Apesar de ser uma etapa demorada e maçante em um aplicativo de grande porte, é essencial para que as pessoas que não o desenvolveram possam obter o completo entendimento do sistema.

Após várias depurações do sistema e conversas informais com os desenvolvedores, foi possível compreender as partes principais, necessárias para a integração do Kinect. Desta forma, foram importadas para o Sistema ITV as bibliotecas: Kinect for Windows SDK e

Win.h, com as funções responsáveis pelo rastreamento do esqueleto e a manipulação do mouse, respectivamente. Além disso, foram criadas duas classes principais para o tratamento dos dados obtidos:

- Esqueleto: Esta classe se integra ao Sistema ITV através de parâmetros em seu método construtor, o que permite a manipulação do sistema através métodos executados por ela. Esta classe contém todas as funções de obtenção e tratamento de dados do Kinect, como reconhecimento de gestos, navegação e simulação do mouse.
- DTW: Esta classe recebe como argumentos os dois padrões a serem comparados e os respectivos limiares de cada eixo. É calculado a semelhança entre as curvas e retornado um valor numérico.

Estas classes representam a ponte de ligação com o Sistema ITV e devem estar em sincronismo para seu perfeito funcionamento. Ainda, possuem documentação básica, explicando a utilidade de seus métodos para futuras melhorias e implementações

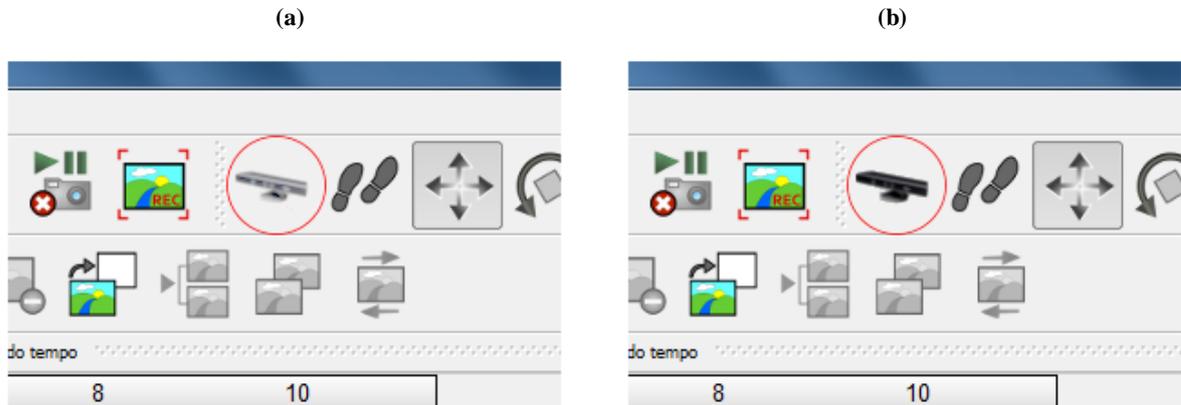
5.4.2 Interface Gráfica

Algumas alterações na *Graphical User Interface* (GUI) do sistema foram necessárias para a implementação do reconhecimento de gestos. O Sistema ITV tem sua interface gráfica desenvolvida através do *framework* Qt (BARATA, 2010). Esta biblioteca possui um intuitivo editor de interfaces gráficas, denominado *Qt Designer*. Através desse editor de GUI, foram feitas as seguintes modificações:

- Adição de um ícone na interface principal para ativar/desativar o Kinect
- Alteração da caixa de dialogo de configuração de passos
- Criação de uma lista de movimentos para transição de passos

O ícone que ativa/desativa o funcionamento do dispositivo é uma imagem que representa o Kinect. Este permanece claro enquanto o dispositivo está inativo e escurece ao ser ativado. A Figura 5.12 apresenta (circulado em vermelho) o ícone em seus dois modos: (a) inativo e (b) ativo.

Figura 5.12 – Ícone do Kinect (a) inativo e (b) ativo.



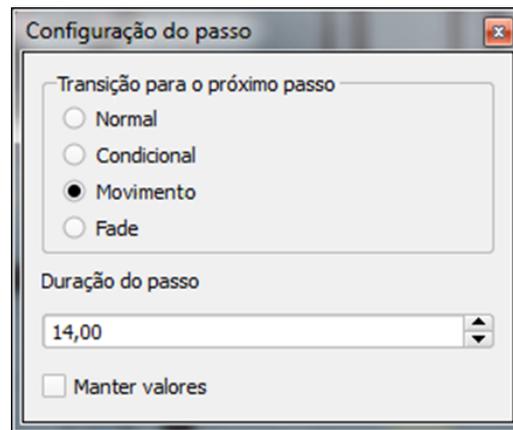
Fonte: Próprio autor.

As animações do Sistema ITV funcionam por meio passos de sequenciais. Antes da modificação feita na caixa de diálogo de configuração de passos, existiam três tipos de transição: Normal, *Fade*, Condicional.

A transição Normal simplesmente avança o passo, sem nenhum efeito gráfico. A transição *Fade* escurece lentamente o fim do passo atual e clareia lentamente o início do passo seguinte, geralmente utilizada em mudanças de ambiente. A transição Condicional se refere ao clique do mouse em algum objeto da cena atual, ou seja, o procedimento só progride se um objeto previamente estabelecido for selecionado, normalmente este tipo de transição é utilizada para selecionar ferramentas ou pressionar botões nas telas de gerenciamento do Sistema Aberto de Supervisão e Controle (SAGE).

Desta forma, foi adicionado um novo tipo de transição, denominado Movimento. A transição de Movimento segue o princípio da transição Condicional, contudo, não é a seleção do objeto que avança o passo, e sim o reconhecimento de um gesto estabelecido. A Figura 5.13 mostra a caixa de diálogo de configuração de passos modificada.

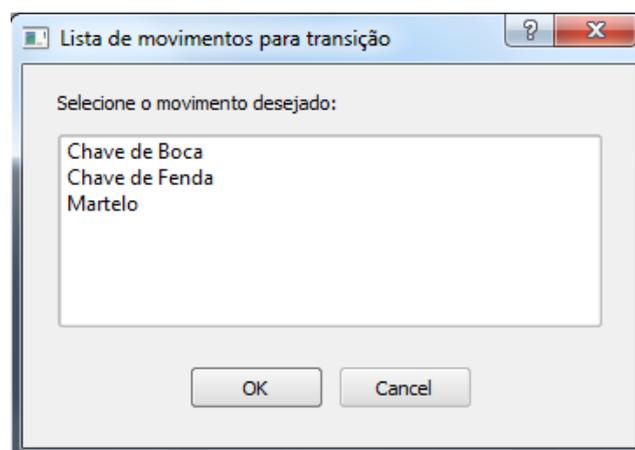
Figura 5.13 – Caixa de diálogo da configuração do passo.



Fonte: Próprio autor.

Ao selecionar o tipo de transição de Movimento, abre-se uma nova caixa de diálogo com uma lista de movimentos para transição. Nessa lista, constam os três gestos estabelecidos anteriormente: Chave de Boca, Chave de Fenda e Martelo. O Sistema ITV, através do sensor Kinect, deve ser capaz reconhecer e validar o movimento selecionado para que o procedimento possa avançar. A Figura 5.14 ilustra a interface da lista de movimentos para transição.

Figura 5.14 - Caixa de diálogo da lista de movimentos para transição.



Fonte: Próprio autor.

5.4.3 Funcionamento

Ao iniciar o Sistema ITV, o sensor Kinect ainda não está ativo. É preciso que o usuário o faça através do ícone do Kinect, que escurece após pressionado. Feito isso, o

sistema passa a rastrear o corpo do indivíduo em seu funcionamento como simulador de mouse, ou seja, no modo controlador de interface.

Se em determinado instante o indivíduo decida pelo modo de navegação, basta erguer os dois braços à altura dos ombros durante 1.5 segundos e, caso decida sair, basta que baixe um dos braços, retornando para o modo controlador de interface.

Os procedimentos executados pela ITV acontecem em passos sequenciais, assim, alguns passos requerem a interação do usuário para sua continuidade. No fim dos passos em que a transição é de Movimento, é necessário que o usuário faça o gesto escolhido para aquele passo a fim que a ITV siga em frente. Neste momento (fim do passo) onde a transição é de Movimento, o sistema sai do modo de controle de interface, entra no modo reconhecedor de gestos e aguarda o gesto ser executado. Após a movimentação correta, o sistema retorna ao controle de interface.

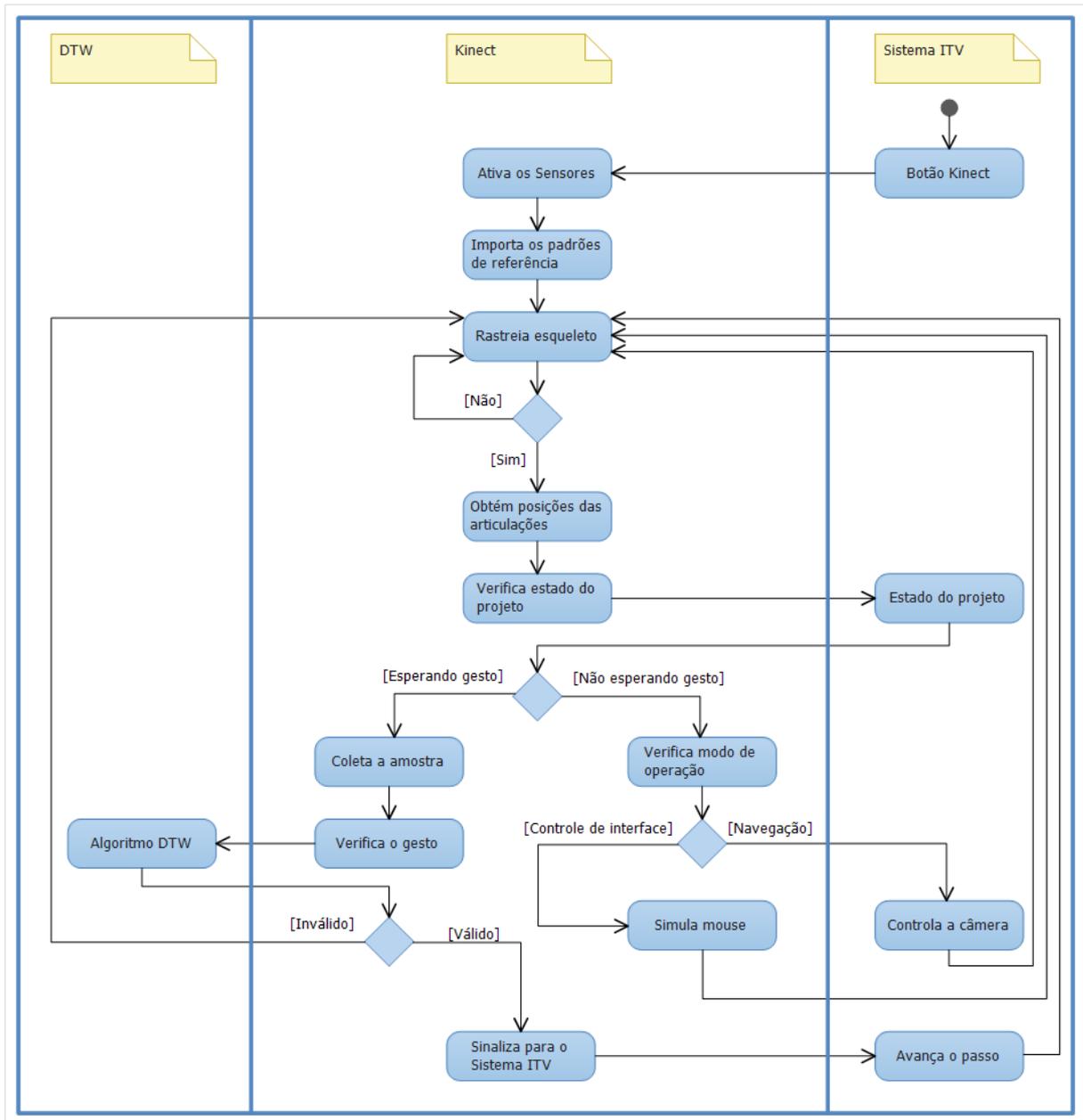
Para isso, foi preciso implementar três métodos: um verificador de tipo de passo, um verificador de tipo de movimento e um sinalizador de gesto reconhecido. Assim, ao fim de cada passo com transição de Movimento, o Sistema ITV aguarda uma resposta do sinalizador de gesto reconhecido.

Foi definido que o gesto deve ter 2 segundos e deve ser executado quando o ícone do Kinect escurecido, ou seja, ativo. Este ícone tem sua aparência alternada de ativo para inativo a cada 2 segundos, durante o reconhecimento de gestos. Além disso, foram implementados alertas sonoros, ou *beeps*, por dois motivos: facilitar a identificação do tempo correto de se executar o gesto e fornecer um *feedback* de erro ou de acerto do gesto. Os *beeps* são uma solução alternativa para o fato de que não existe nenhum objeto ou controle junto ao corpo, o que poderia resultar em um *feedback* oriundo deste próprio objeto.

No início do reconhecimento, ou seja, no tempo 0 segundo, o alerta sonoro é um som com uma frequência de 1000 Hz durante 0.1 segundos. No fim do reconhecimento, ou seja, no tempo igual a 2 segundos, existem duas possibilidades: caso o gesto seja considerado correto, o sistema emite um *beep* igual de 1000 Hz por 0.05 segundos e, caso o gesto seja inválido, é emitido um alerta sonoro mais grave com uma frequência de 400 Hz durante 0.05 segundos.

Para melhor compreender o funcionamento interno do sistema e as novas funções implementadas, a Figura 5.15 apresenta um diagrama de atividades da integração do Kinect ao Sistema ITV.

Figura 5.15 - Diagrama de atividades.



Fonte: Próprio autor.

6 ESTUDO DE CASO

Nesse capítulo será mostrada, como estudo de caso, uma instrução técnica virtual elaborada com o objetivo de testar e avaliar a navegação, o controle da interface e o reconhecimento de gestos através do sensor Kinect.

6.1 ITV DE TESTE E AVALIAÇÃO

Esta ITV foi composta de poucos passos, porém, que fizessem uso de transições condicionais e de movimento, para que o usuário utilizasse o Kinect simulando o mouse e simulando movimentos das ferramentas. Além disso, foram carregados todos os modelos de uma subestação elétrica para que o indivíduo pudesse navegar livremente e conhecer melhor o ambiente virtual. A Figura 6.1 ilustra a navegação pelo ambiente virtual através do Kinect.

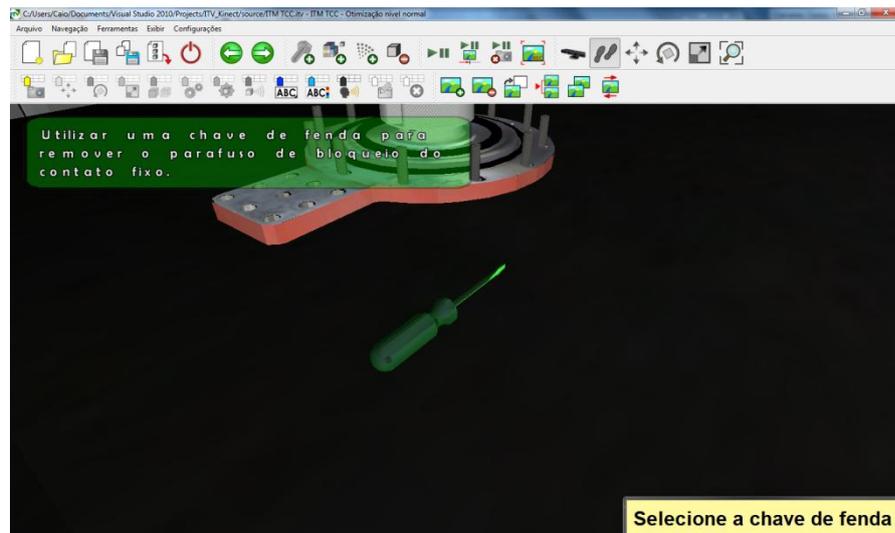
Figura 6.1 – Navegação no ambiente através do Kinect.



Fonte: Próprio autor.

A Figura 6.2 expõe um passo Condicional onde é necessário que o usuário selecione uma chave de fenda para prosseguir com o procedimento.

Figura 6.2 - Passo com transição Condicional de seleção da chave de fenda.



Fonte: Próprio autor.

Este passo requer a seleção da chave de fenda com o cursor do mouse para que o procedimento siga adiante. Através do controle de interface implementado, é possível selecionar a chave de fenda por meio de movimentos com as mãos direita e esquerda. A Figura 6.3 mostra o usuário utilizando o controle de interface para a seleção da chave de fenda.

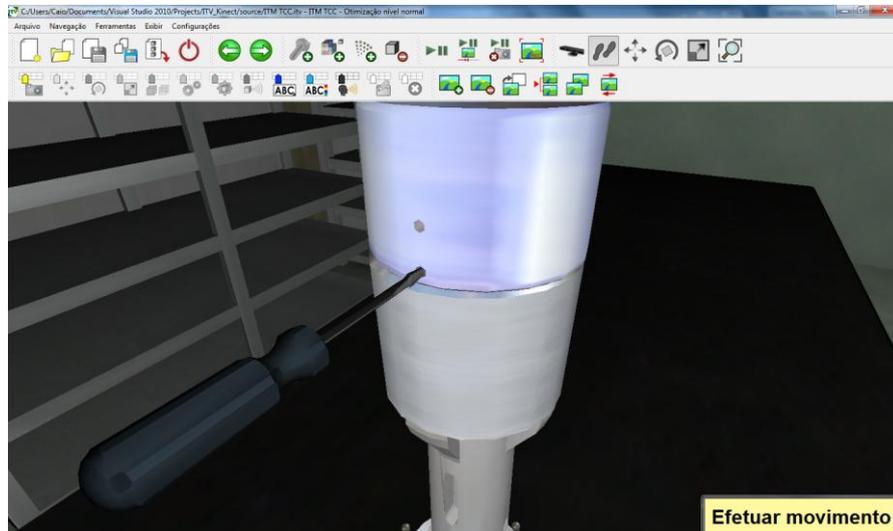
Figura 6.3 – Controle de interface para a seleção da chave de fenda.



Fonte: Próprio autor.

A Figura 6.4 mostra um passo com transição de Movimento, onde é preciso que o usuário faça o gesto de utilização de chave de fenda para que seja dada continuidade à ITV.

Figura 6.4 - Passo com transição de Movimento de chave de fenda.



Fonte: Próprio autor.

A Figura 6.5 exibe o usuário simulando o movimento de chave de fenda.

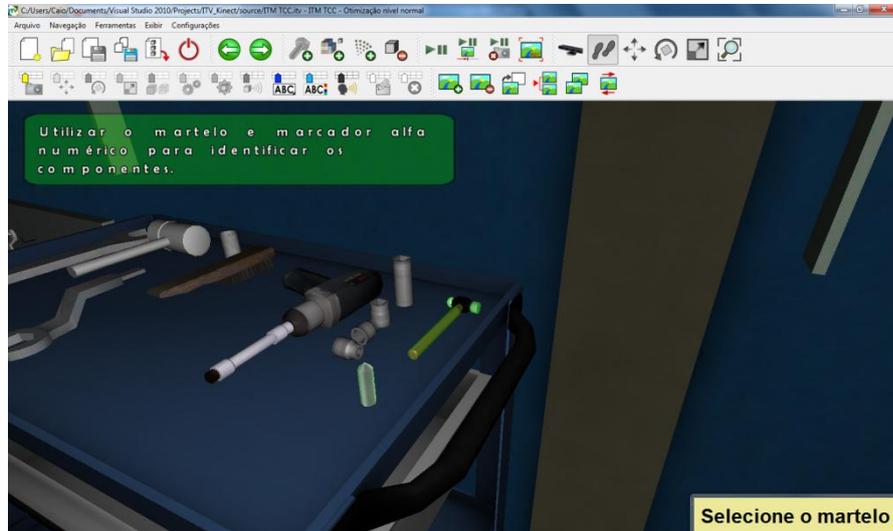
Figura 6.5 – Simulação de movimento de chave de fenda.



Fonte: Próprio autor.

A Figura 6.6 mostra um passo Condicional onde é necessário que o usuário selecione o martelo para que a ITV siga em frente.

Figura 6.6 – Passo com transição Condicional de seleção de martelo.



Fonte: Próprio autor.

A Figura 6.7 apresenta o usuário utilizando o controle de interface através o Kinect no passo de transição Condicional de seleção do martelo.

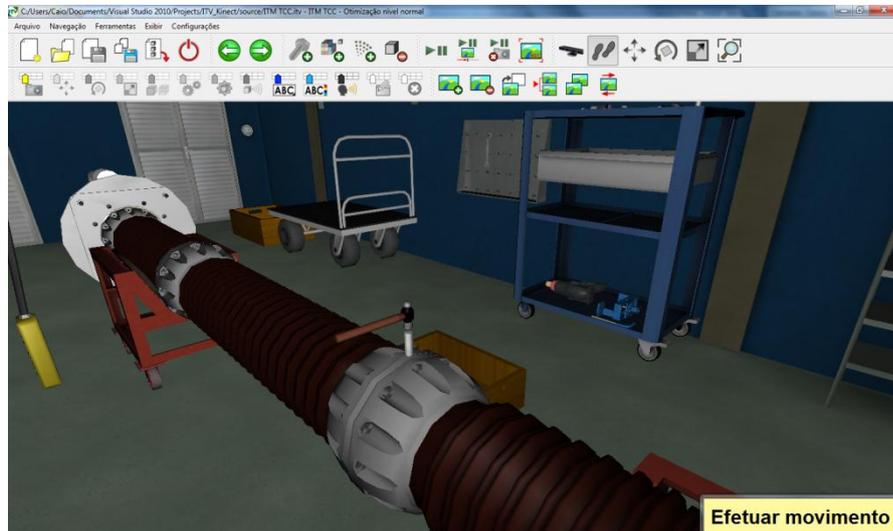
Figura 6.7 - Controle de interface para a seleção do martelo.



Fonte: Próprio autor.

A Figura 6.8 exibe o passo de transição de Movimento, onde é necessária a movimentação de martelo para que a ITV progrida.

Figura 6.8 – Passo com transição de Movimento de martelo.



Fonte: Próprio autor.

A Figura 6.9 exibe o usuário simulando a movimentação de martelo para que o sistema possa reconhecer como gesto válido e seguir em frente na Instrução Técnica Virtual.

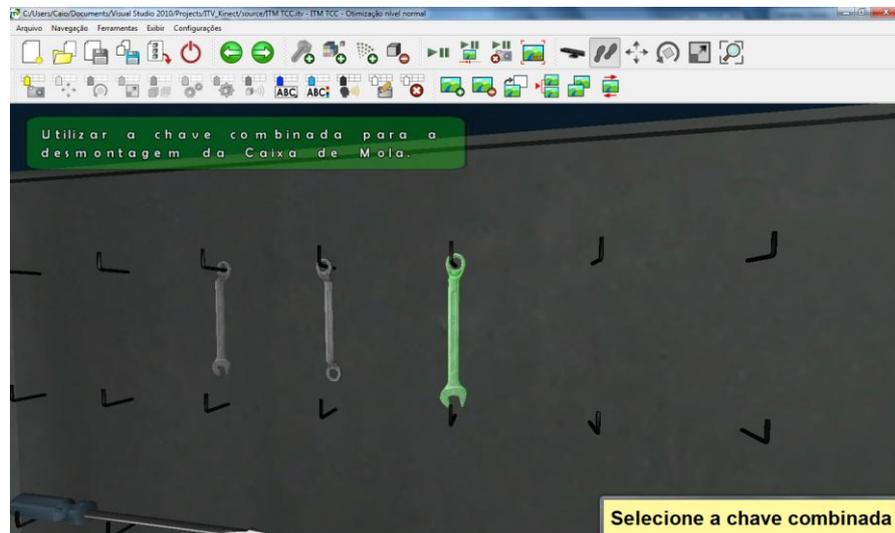
Figura 6.9 - Simulação de movimento de martelo.



Fonte: Próprio autor.

A Figura 6.10 mostra um passo Condicional para a seleção da chave combinada para o processo de desmontagem da Caixa de Mola.

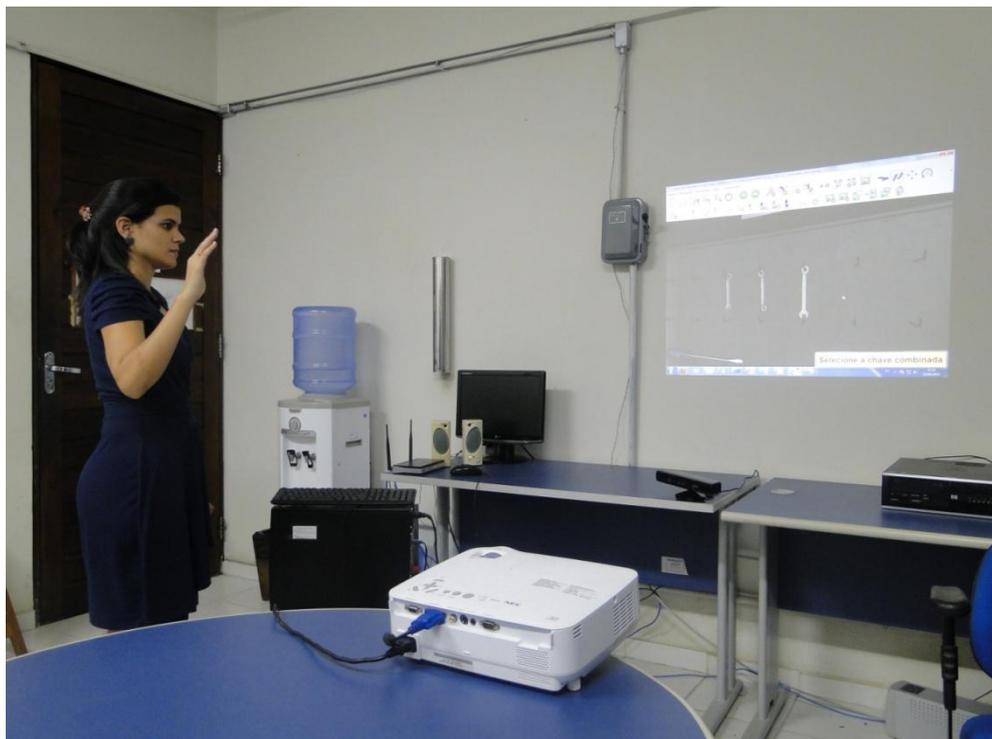
Figura 6.10 – Passo com transição Condicional de seleção de chave combinada.



Fonte: Próprio autor.

A Figura 6.11 exibe o usuário utilizando o controle de interface pelo Kinect para selecionar a chave combinada.

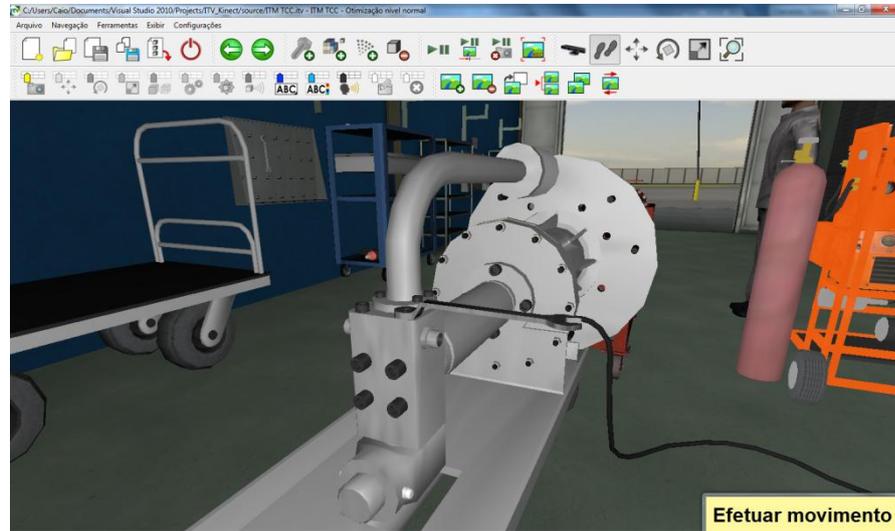
Figura 6.11 – Controle de interface para a seleção da chave combinada.



Fonte: Próprio autor.

A Figura 6.12 ilustra o passo com transição de Movimento de chave de boca (ou combinada).

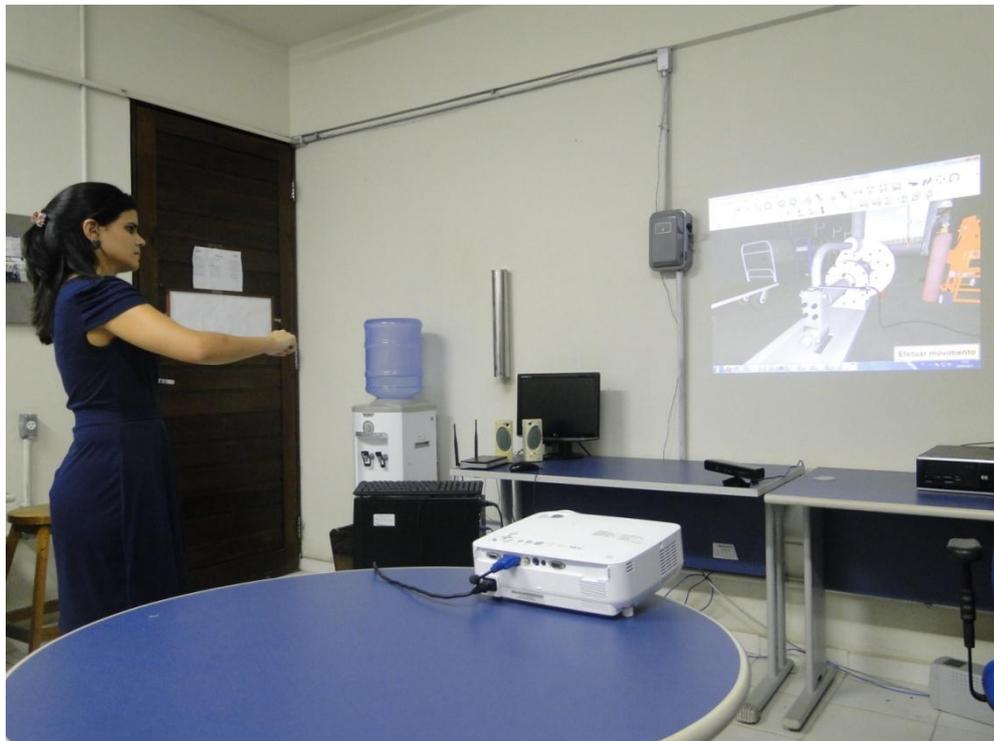
Figura 6.12 – Passo com transição de Movimento de chave de boca.



Fonte: Próprio autor.

A Figura 6.13 mostra o usuário executando o movimento de chave de boca.

Figura 6.13 – Simulação de movimento de chave de boca.



Fonte: Próprio autor.

Vale ressaltar que a utilização do Kinect é, na maioria das vezes, para instruções técnicas de manutenção e não instruções técnicas de operação. Isso se dá por não haver a necessidade de simular o mouse e o teclado quando, na própria empresa, o procedimento é feito no computador, através do SAGE e de outros sistemas gerenciadores.

7 AVALIAÇÃO

Neste capítulo serão mostrados os resultados obtidos em uma pesquisa de satisfação sobre o sensor Kinect, como interface de entrada, no Sistema ITV. Com essa avaliação, espera-se quantificar a aceitação ou rejeição deste trabalho. Esta pesquisa foi realizada com treze estudantes, de 20 à 28 anos, da Universidade Federal do Pará, ou seja, não são usuários finais do Sistema ITV.

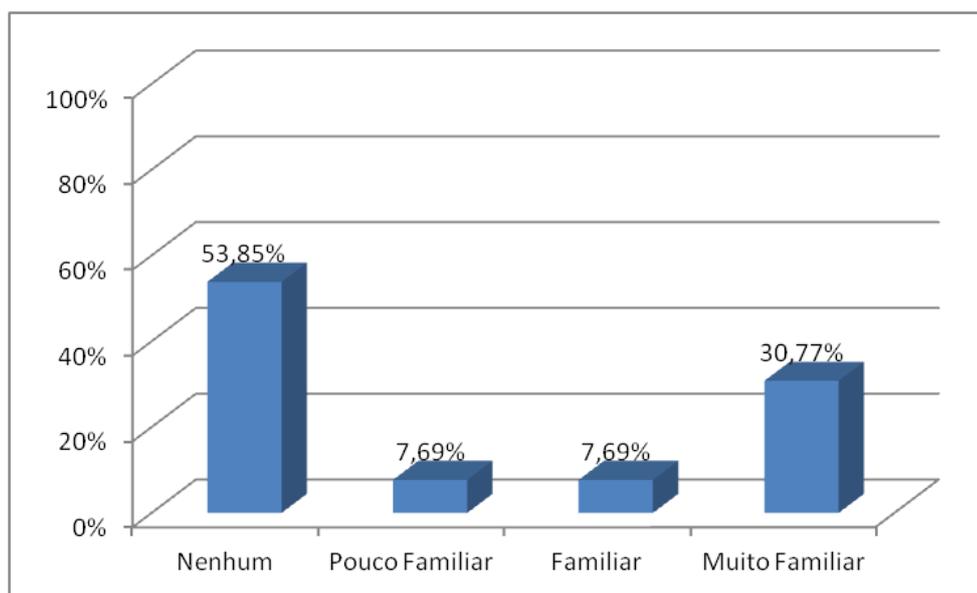
Os entrevistados realizaram a navegação pelo ambiente da subestação elétrica e, em seguida, seguiram os procedimentos da ITV criada para testar e avaliar o dispositivo. Esta ITV foi detalhada no capítulo 6.

Neste caso, os estudantes responderam o questionário que consta no Apêndice A desta dissertação. Este questionário é constituído por 7 questões com respostas de múltipla escolha e, no fim, uma campo para observações diversas.

7.1 RESULTADOS

A primeira questão tratava do grau de familiaridade com o Sistema ITV. A Figura 7.1 exibe os resultados obtidos.

Figura 7.1 – Grau de familiaridade com o Sistema ITV.

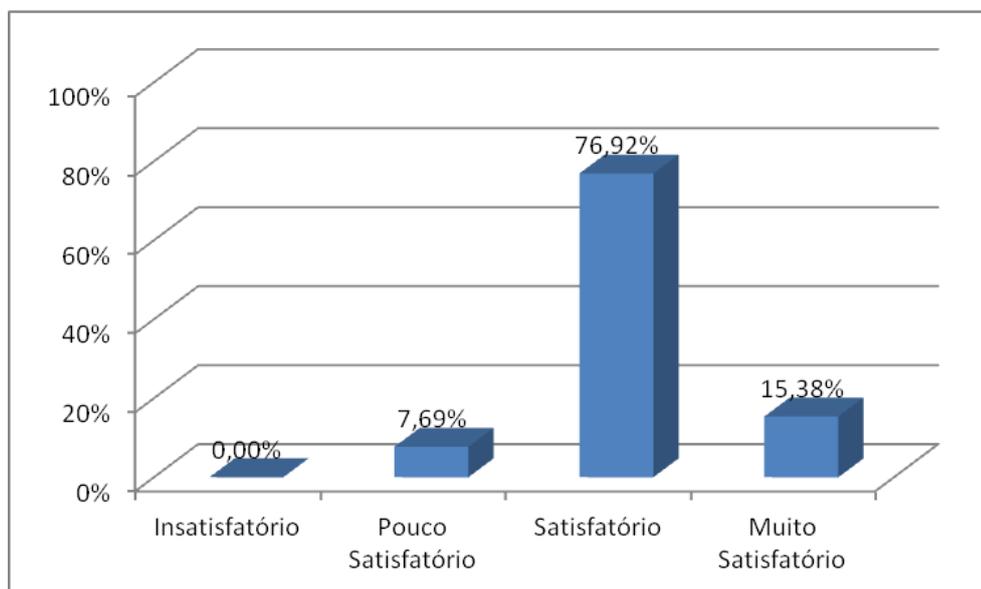


Fonte: Próprio autor.

Dentre os estudantes avaliados, uma parte apresentava familiaridade com o Sistema ITV e outra parte o desconhecia. Pode-se perceber que mais da metade dos entrevistados (53.85%) não possuía familiaridade alguma com o sistema.

Para descobrir a satisfação dos usuários em relação ao controle de interface, foi feita a seguinte questão: “Avalie o controle de interface do Sistema ITV através do Kinect.”. A Figura 7.2 exibe os resultados obtidos da satisfação sobre o controle de interface.

Figura 7.2 – Satisfação sobre o controle de interface através do Kinect.

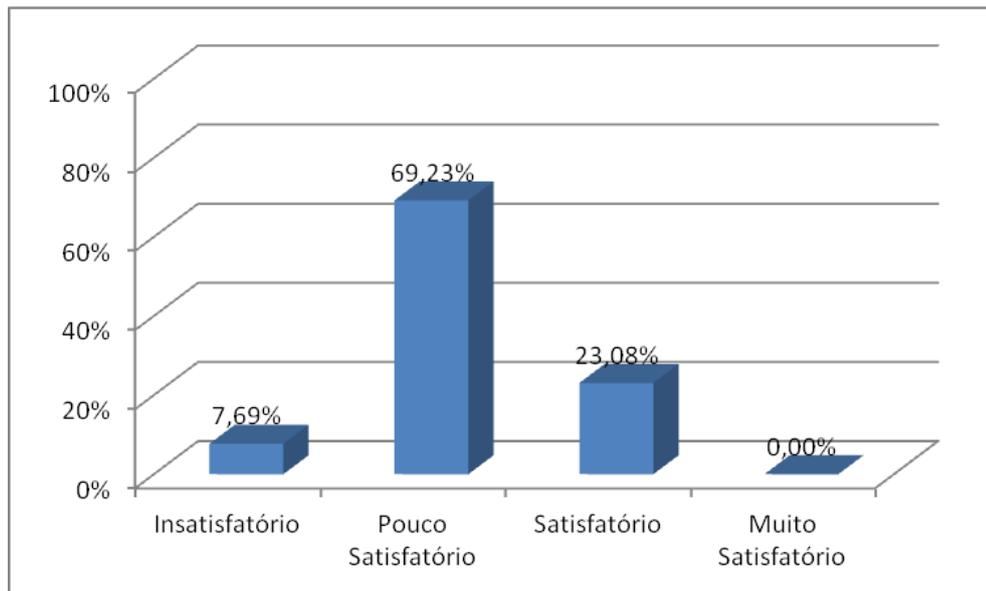


Fonte: Próprio autor.

É possível notar que o controle da interface foi bem aceito, com 92.3% de satisfação. Deste modo, é possível considerar que o processo de adaptação para o controle do mouse, através do Kinect, não foi uma dificuldade.

A terceira questão abordava a satisfação em relação à navegação, da seguinte forma: “Avalie a navegação no ambiente virtual do Sistema ITV através do Kinect.”. A Figura 7.3 ilustra os resultados obtidos pela pesquisa sobre a navegação.

Figura 7.3 – Satisfação sobre a navegação no ambiente através do Kinect.

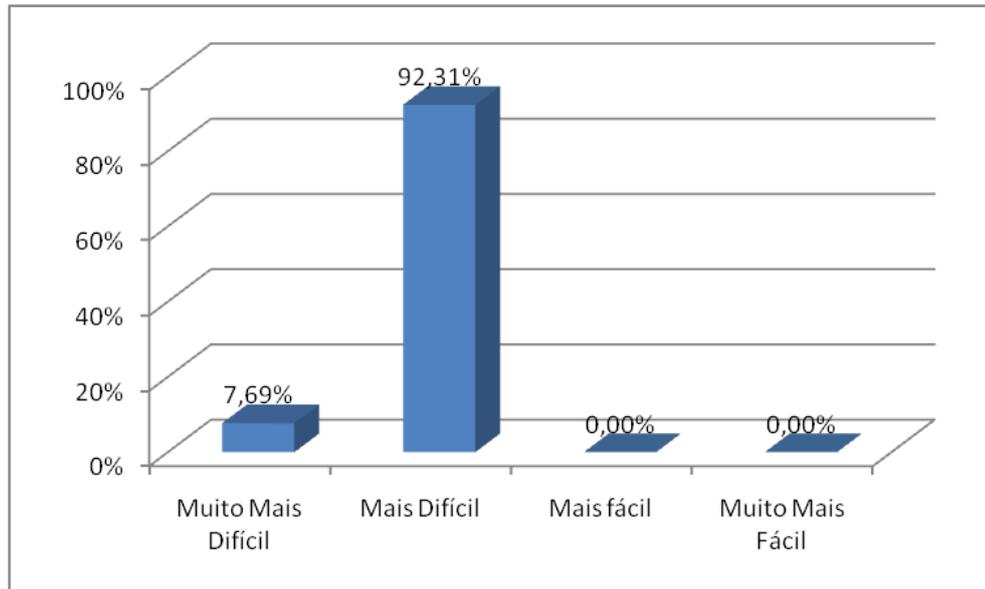


Fonte: Próprio autor.

A navegação foi considerada majoritariamente pouco satisfatória (69.23%), além de 7.69% dos entrevistados a avaliarem como insatisfatória. Portanto, conclui-se que é preciso modificações no projeto para uma melhor aceitação da interface, neste aspecto. Contudo, foi observado que a curva de aprendizado é lenta, ou seja, é preciso certo tempo para se acostumar com o sistema de navegação. Os estudantes tiveram pouco tempo para adaptação e este fato pode ter contribuído para rejeição da navegação por meio do Kinect.

Com a intenção de averiguar o grau de adaptação dos usuários ao sistema de navegação, foi feita a seguinte questão: “Comparado ao mouse e ao teclado, a navegação no ambiente virtual através do Kinect é:”. A Figura 7.4 mostra os resultados obtidos sobre a utilização do Kinect em relação ao mouse e teclado.

Figura 7.4 – Resultados sobre o Kinect em relação ao mouse e teclado.

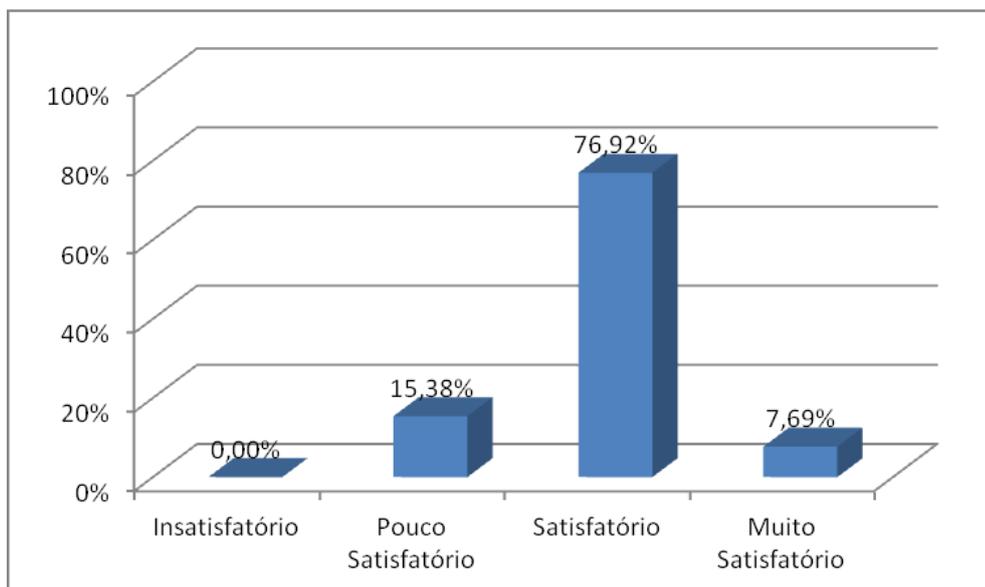


Fonte: Próprio autor.

Os dados colhidos foram conclusivos e mostraram que a adaptação foi mais difícil para 92.31% e muito mais difícil para 7.69% dos estudantes. Porém, vale ressaltar que mouse e teclado são interfaces padrão, ou seja, a maioria das pessoas já está acostumada.

Para descobrir a satisfação dos estudantes em relação ao reconhecimento de gestos, a quinta questão foi a seguinte: “Avalie o reconhecimento de gestos no Sistema ITV.”. A Figura 7.5 exibe os resultados da satisfação sobre o reconhecimento de gestos.

Figura 7.5 – Satisfação sobre o reconhecimento de gestos através do Kinect.

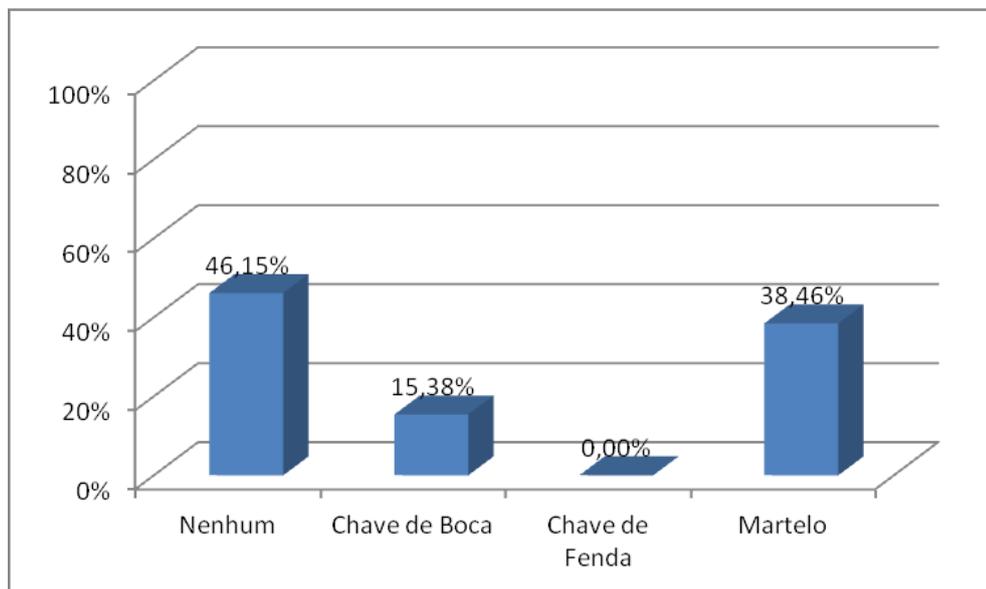


Fonte: Próprio autor.

Os resultados obtidos mostraram que o reconhecimento de gestos foi avaliado como satisfatório ou muito satisfatório por 84.61% dos entrevistados. Assim, pode-se considerar que a metodologia utilizada no reconhecimento dos gestos apresentou uma boa aceitação.

A sexta questão tratava da dificuldade no reconhecimento de algum gesto específico, com a questão: “Você teve mais dificuldade no reconhecimento de algum gesto específico? Qual?”. A Figura 7.6 ilustra a dificuldade em gestos específicos.

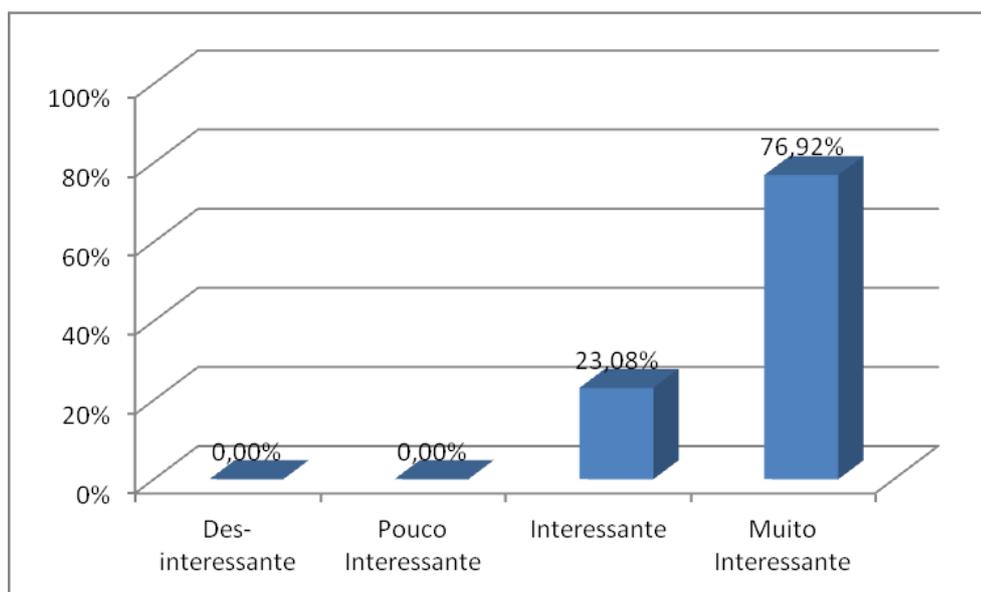
Figura 7.6 – Dificuldade no reconhecimento dos gestos.



Fonte: Próprio autor.

É possível notar que praticamente a metade dos entrevistados (46.15%) não apresentou dificuldades nos gestos. Nenhum estudante apresentou dificuldade na movimentação de chave de fenda. Apenas 15.38% tiveram dificuldade com o gesto de chave de boca. Porém, um número relativamente alto de pessoas (38.46%) apresentou dificuldades com a movimentação de martelo. Pode ser preciso reavaliar o padrão de referência e etapas de coletas de amostras, determinação de limiares e testes de validação para que este gesto de martelo se torne menos difícil.

A última questão tratava do grau de atratividade do aprendizado através da interface proposta com a seguinte pergunta: “Em sua opinião, quão interessante torna-se o aprendizado mediante a interação com Sistema ITV através do Kinect?”. A Figura 7.7 mostra o resultado obtido.

Figura 7.7 – Grau de atratividade do aprendizado através do Kinect.

Fonte: Próprio autor.

Este resultado foi conclusivo e 100% dos entrevistados avaliaram o aprendizado com sensor Kinect no Sistema ITV como interessante ou muito interessante. Portanto, concluí-se que este trabalho desperta o interesse no aprendizado.

Com uma ferramenta atrativa, é possível que o usuário aumente sua concentração na realização de tarefas. Este aspecto é importante em treinamentos virtuais, pois o usuário deve se concentrar nos deveres para aprendê-los de forma correta.

Observações e sugestões interessantes foram feitas pelos estudantes. Uma dessas observações foi a dificuldade de identificar como é o gesto requisitado pelo sistema, ou seja, qual a movimentação correta para que o sistema reconheça. Outra dificuldade encontrada foi a identificação da região de rastreamento das mãos na navegação. Uma das sugestões apresentadas foi a adição da representação dos movimentos pelo avatar, o que tornaria a interação com o ambiente mais envolvente.

Estas observações e sugestões serão comentadas no capítulo que conclui este trabalho.

8 CONCLUSÃO E TRABALHOS FUTUROS

Esta dissertação apresentou a implementação do dispositivo Kinect em um sistema de treinamento virtual para operadores e mantenedores de hidrelétricas e subestações elétricas, denominado Sistema ITV. Essa integração entre dispositivo e sistema permitiu o controle de interface, navegação no ambiente virtual e reconhecimento de gestos pré-estabelecidos.

Procurou-se fundamentar conceitos sobre as tecnologias envolvidas e apresentar trabalhos semelhantes a este, bem como mostrar a importância dessas tecnologias na busca de novas formas de interação com o computador. Foi abordado com mais detalhes o Sistema ITV e apontado os aperfeiçoamentos recentes. Em seguida, foi explicado o funcionamento do sensor Kinect, suas vantagens e desvantagens em relação a outras NUI's, além de introduzir conceitos e definições matemáticas sobre o algoritmo de comparação de curvas, o DTW.

No desenvolvimento da dissertação, foram detalhados o projeto e implementação da integração entre o Kinect e o Sistema ITV em seus três modos de funcionamento. Como resultado prático, foi mostrada a execução de uma Instrução Técnica Virtual (ITV), elaborada especialmente para testar e avaliar a interação com o sistema por meio do Kinect. Por fim, foram apresentados resultados obtidos através de pesquisa de satisfação, realizada com treze estudantes da UFPA. Esses resultados foram apresentados em forma de gráficos quantitativos, esclarecendo melhor os dados obtidos.

Os objetivos deste trabalho foram alcançados com sucesso, pois foram implementados o controle da interface, a navegação pelo ambiente e o reconhecimento de três gestos através do sensor Kinect. Além disso, os resultados obtidos foram, em geral, considerados satisfatórios. Contudo, ajustes e melhorias precisam ser realizados até o ponto de sua versão final estabilizada.

Em comparação com Fonseca (2013), os resultados alcançados foram aproximados, pois o trabalho com o *Wii Remote Controller* atingiu um índice de 84% de acerto na movimentação de martelo, 72% no giro anti-horário e 92% no giro horário. Já o trabalho desta dissertação alcançou um índice de 86.67% na movimentação de martelo e 73.33% na movimentação de chave de fenda, equivalente aos giros horários e anti-horários mencionados.

Um dos problemas enfrentados foi o retorno de falso-positivos na movimentação da chave de fenda, pois como explicado, o Kinect não reconhece a rotação da mão, somente a

posição. Outra dificuldade encontrada no reconhecimento de gestos, assim como Doliotis *et al* (2011) observou, foi o ponto de vista da câmera do Kinect, onde é necessário que o indivíduo esteja sempre de frente para que o rastreamento de sua posição seja ótimo.

Ao integrar o Kinect ao Sistema ITV, uma enorme quantidade de implementações é possível. Até então, o Sistema ITV conta com o reconhecimento de três gestos, porém, esta quantidade pode ser ampliada com uma forma de criação automática de gestos dentro do próprio sistema. Deste modo, o usuário não necessitaria ter o conhecimento de programação para criar um novo gesto a ser reconhecido.

Pode-se implementar, também, um modo para a inversão das mãos no rastreamento e no reconhecimento de gestos, ou seja, um modo para canhotos, o que facilitaria o controle de interface e o reconhecimento de gestos para pessoas que estão habituadas a executarem tarefas com a mão esquerda.

Para as observações apresentadas pelos estudantes avaliados, como a dificuldade de identificar a movimentação correta do gesto requisitado e a dificuldade de identificação da região de movimentação das mãos, propõe-se um treinamento básico de utilização da nova interface.

Uma das sugestões oferecidas pelos entrevistados está relacionada à representação dos gestos, dentro do Sistema ITV, através de animações do avatar, ou seja, a representação humana no ambiente virtual executaria o mesmo gesto que foi reconhecido pelo sistema. Esta representação pode tornar o ambiente virtual mais realístico. Para isso, é necessária a animação de cada movimento no *software* de modelagem e, em seguida, o carregamento do avatar animado no sistema.

Por fim, uma melhoria significativa a ser investigada é o desenvolvimento de uma metodologia de reconhecimento de gestos que não necessitasse de um tempo fixo para a realização do movimento, ou seja, o usuário poderia fazer o gesto a qualquer momento (no final do passo com transição de Movimento) e o sistema se encarregaria de detectar o início e o fim do movimento automaticamente.

Conclui-se que as vantagens do sensor Kinect, como *Natural User Interface*, em treinamentos virtuais podem compensar suas desvantagens. A integração entre o Kinect e o Sistema ITV tornou o aprendizado mais dinâmico e interessante, desta forma, os usuários se concentram mais para realizar as tarefas e, conseqüentemente, podem aprender de maneira rápida e correta.

REFERÊNCIAS

ALMEIDA, F. A. M. **Classificação Automática de Veículos pelo Perfil Magnético Através de Técnicas de Aprendizagem de Máquina**. Dissertação (Mestrado Profissional em Computação Aplicada) - Universidade Estadual do Ceará, Fortaleza, CE, 2010.

ARROYO, E.; ARCOS, L. L. J. **SRV: A Virtual Reality Application to Electrical Substations Operation Training**. In: IEEE International Conference on Multimedia Computing and Systems 1999 (ICMCS 99), v. 1, p. 835-839, Florence, Italy, 1999. DOI: 10.1109/MMCS.1999.779309.

BARATA, P. N. A. **Projeto e Implementação de um Sistema de Autoria para Animações, Simulações e Treinamentos em Realidade Virtual**. Dissertação (Mestrado em Engenharia Elétrica) - Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Pará, Belém, PA, 2010.

BARTOLI, G. *et al.* **Emergency Medicine Training with Gesture Driven Interactive 3D simulations**. In: Proceedings of the 2012 ACM workshop on User experience in e-learning and augmented technologies in education (UXeLATE '12), p. 25-30, New York, USA, 2012. ISBN: 978-1-4503-1593-7 DOI: 10.1145/2390895.2390903

BELLMAN, R; KABALA, R. On Adaptive Control Processes. **IRE Transactions on Automatic Control**, v. 4, no. 2, p. 1-9, 1959. ISSN: 0096-199X. DOI: 10.1109/TAC.1959.1104847.

BLANCHETTE, J.; SUMMERFIELD, M. **C++ GUI Programming with Qt 4**. 2ª ed. USA: Prentice Hall, 2008.

BLUEMEL, E. *et al.* **Virtual Environments for the Training of Maintenance and Service Tasks**. In: Proceedings of the 2003 Winter Simulation Conference (WSC 2003), v. 2, p. 2001-2007, New Orleans, EUA, 2003.

BLUEMEL, E. *et al.* **Technology Enhanced Training at Workplace: A Virtual Reality Based Training System for the Technical Domain**. In: 1st International Conference on EBusiness and E-Learning Proceedings, pp. 57-62, 2005.

CHIAVENATO, I. **Recursos Humanos**. 2ª ed. pp. 288, São Paulo: Atlas, 1985.

CLINTON, Y. **Game Character: Modeling and Animation with 3ds Max**. USA: Focal Press, 2007. ISBN-10: 0240809785.

DEITEL, H.; DEITEL, P. **C++ How to Program**. 6^a ed., USA: Prentice Hall, 2009.

DOLIOTIS, P. *et al.* **Comparing Gesture Recognition Accuracy Using Color and Depth Information**. In: Proceedings of the 4th International Conference on Pervasive Technologies Related to Assistive Environments (PETRA '11), n. 20, New York, USA, 2011. ISBN: 978-1-4503-0772-7. DOI: 10.1145/2141622.2141647.

FFMPEG. FFmpeg. Disponível em: <<http://www.ffmpeg.org>> Acesso em: 22 abr. 2013.

FRANCESE, R.; PASSERO, I.; TORTORA, G. **Wiimote and Kinect: gestural user interfaces add a natural third dimension to HCI**. In: Proceedings of the International Working Conference on Advanced Visual Interfaces 2012 (AVI '12), p. 116-123, New York, USA, 2012. ISBN: 978-1-4503-1287-5. DOI: 10.1145/2254556.2254580.

FMOD. **Fmod Interactive Audio**. Disponível em: <<http://www.fmod.org>>. Acesso em: 12 jul. 2012.

FONSECA, L. M. **Implementação da Detecção de Movimentos, Navegação e Controle da Interface do Sistema ITV através do Dispositivo Wii Remote Controller**. Trabalho de Conclusão de Curso (Graduação em Engenharia da Computação) - Faculdade de Engenharia da Computação, UFPA, Belém, PA, 2013.

GIMP. The GNU Image Manipulation Program. Disponível em: <<http://www.gimp.org>> Acesso em: 22 jun. 2012.

GONÇALVES, N. *et al.* **Automatic Detection of Stereotypical Motor Movements**. In: 26th European Conference on Solid-State Transducers (EUROSENSOR 2012), v. 47, p. 590–593, 2012. DOI: 10.1016/j.proeng.2012.09.216

HELANDER, M. G.; LANDAUER, T. K.; PRABHU, P. V. **Handbook of Human-Computer Interaction**. 2^a ed. North-Holland: Elsevier Science B.V., 1997. ISBN: 0-444-81862 6

IBERDROLA. Iberdrola. Disponível em: <<http://www.iberdrola.es>> Acesso em: 04 jul. 2012.

JAIN, J.; LUND, A.; WIXON, D. **The Future of Natural User Interfaces**. In: CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11). p. 211-214, New York, USA, 2011. ISBN: 978-1-4503-0268-5. DOI: 10.1145/1979742.1979527

JAVA. Java SE Desktop Technologies - Java 3D API. Disponível em: <<http://www.oracle.com/technetwork/java/javase/tech/index-jsp-138252.html>> Acessado em: 08 ago. 2012.

JEHAN, T. **Creating Music By Listening**. Tese (Doutorado em Filosofia) - Program in Media Arts and Sciences, Massachusetts Institute of Technology, Cambridge, MA, EUA, 2005.

JUNIOR, A. P. *et al.* **Instruções Técnicas De Manutenção E Operação Virtuais De Uma Unidade Geradora Hidráulica**. In: XX Seminário Nacional De Produção e Transmissão De Energia Elétrica (SNPTEE 2009), Recife, PE, 2009.

JUNIOR, A. P. **Hidrogeradora Virtual: Utilizando Técnicas de Realidade Virtual Desktop para o Estudo de uma unidade Hidrelétrica de Energia**. Dissertação (Mestrado em Engenharia Elétrica) - Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Pará, Belém, PA, 2006.

JUNKER, G. **Pro Ogre 3D Programming**, [S.l.]: Apress, 2006.

KEOGH, E. J.; PAZZANI, M. J. **Scaling up Dynamic Time Warping to Massive Datasets**, In: 3rd European Conference on Principles of Data Mining and Knowledge Discovery (PKDD '99), p. 1-11, Prague, Czech Republic, 1999. ISBN:3-540-66490-4.

KINECT. Kinect for Windows SDK - Documentation. Disponível em: <<http://msdn.microsoft.com/en-us/library/hh855347.aspx>> Acesso em: 07 out. 2012.

KINECT. Kinect for Windows - Human Interaction Guideline v1.7. Disponível em: <<http://go.microsoft.com/fwlink/?LinkID=247735>> Acesso em: 22 mar. 2013.

KIRNER, C.; TORI, R. **Realidade Virtual: Conceitos e Tendências**. In: VII Symposium on Virtual Reality (SVR 2004), São Paulo, SP, 2004.

LIMA, L. *et al.* **Um Protótipo para Simulação de Exame de Punção de Mama Utilizando Realidade Virtual Não Imersiva**. In: VII Symposium on Virtual Reality (SVR 2004), São Paulo, SP, 2004.

LOPES, E. S. *et al.* Avaliação do Treinamento de Operadores de *Harvester* com uso de Simulador de Realidade Virtual. **Árvore**, v. 32, n. 2, Viçosa, MG, mar/ago, 2008.

LOPES, E. S.; OLIVEIRA, D.; SILVA, P. C. Avaliação do Desempenho de Operadores no Treinamento com Simulador de Realidade Virtual *Forwarder*. **Ciência Florestal**, v. 20, n. 1, Porto Alegre, RS, jan/mar, 2010. ISSN: 1980-5098 DOI: 10.5902/19805098

MACHADO, L. S.; MORAES, R. M. **Intelligent Decision Making in Training Based on Virtual Reality**. In: Da Ruan. (Org.). Computational Intelligence in complex decision systems. v. 2, p. 85-123, Amsterdam: Atlantis Press, 2009.

MILKOVICH, G. T. **Administração de Recursos Humanos: Treinamento**. São Paulo: Editora Atlas, 2000.

MOTA, E. M. **Movimentação e Ensino de um Robô NAO**. Dissertação (Mestrado Integrado em Engenharia Eletrotécnica e de Computadores) - Faculdade de Engenharia, Universidade do Porto, Porto, Portugal, 2011.

MYSQL. MySQL - The world's most popular open source database. Disponível em: <<http://www.mysql.com>> Acesso em: 08 ago. 2012.

MURDOCK, K. L. **3DS Max 9 Bible**, [S.l.]: Wiley Publishing, 2007.

NORMAN, D. A. Natural User Interfaces are not Natural. **Interactions**. v. 17, n. 3, p. 6-10, New York, NY, USA, mai/jun, 2010. DOI: 10.1145/1744161.1744163

NORMAN, D. A. THE WAY I SEE IT: Simplicity is not the answer. **Interactions - We must redesign professional design education for the 21st century**. v. 15, n. 5, p. 45-46, New York, NY, USA, set/out, 2008. DOI: 10.1145/1390085.1390094

NIELS, R. **Dynamic Time Warping: An intuitive way of handwriting recognition?**. Dissertação de Mestrado, Universidade Radboud, Nijmegen, Holanda, 2004.

NINTENDO. Wii Official Site - What is Wii?. Disponível em: <<http://www.nintendo.com/wii/what-is-wii>> Acesso em: 20 de Março de 2013.

OFUSION. oFusion Technologies. Disponível em: <<http://ofusiontechnologies.com>> Acesso em: 22 de Junho de 2012.

OPENKINECT. OpenKinect, 2012. Disponível em: <<http://www.openkinect.org>> Acesso em: 20 nov. 2012.

PRESSMAN, R. S. **Engenharia de Software**. 6ª ed., [S.l.]: McGraw-Hill, 2006. ISBN: 8586804576.

QUINTANA, J.; MENDONZA E. **3D Virtual Models Applied in Power Substation Projects**. In: 15th International Conference on Intelligent System Applications to Power Systems (ISAP 2009), p. 1-3, Curitiba, Brasil, 2009. DOI: 10.1109/ISAP.2009.5352921.

REZENDE, D. A. **Engenharia de Software e Sistema de Informação**. 3ª ed, Rio de Janeiro: Brasport Livros e Multimídia Ltda, 2005. ISBN: 85-7452-215-5.

SAGE. SAGE - Sistema Aberto de Supervisão e Controle. Disponível em: <<http://www.sage.cepel.br>> Acesso em: 22 abr. 2012.

SENIN, P. **Dynamic Time Warping Algorithm Review**. Information and Computer Science Department, University of Hawaii, pp. 1-23, 2008. Disponível em: <<http://csdl.ics.hawaii.edu/techreports/08-04/08-04.pdf>>

SILVA, E.; MIRANDA, J.; HOUNSELL, M. **Diferenças entre Educação e Treinamento em Ambientes Virtuais 3D**. In: IX Symposium on Virtual and Augmented Reality (SVR 2007), Petrópolis, RJ, 2007.

TORI, R.; KIRNER, C.; SISCOOTTO, R. **Fundamentos e Tecnologia de Realidade Virtual e Aumentada**. VIII Symposium on Virtual Reality (SVR 2006), Belém, PA, 2006.

VILLAROMAN, N.; ROWE, D.; SWAN, B. **Teaching natural user interaction using OpenNI and the Microsoft Kinect sensor**. In: Proceedings of the 2011 conference on Information technology education (SIGITE '11). pp. 227-232, New York, NY, USA, 2011. ISBN: 978-1-4503-1017-8. DOI: 10.1145/2047594.2047654.

VOLPE, R. A. **A Importância do Treinamento para o Desenvolvimento do Trabalho. O Portal dos Psicólogos**, 2009. Disponível em: <<http://www.psicologia.pt/artigos/textos/TL0136.pdf>> Acesso em: 10 mar. 2013.

WIKIMEDIA. Wikimedia Commons - Euclidian vs DTW. Disponível em: <http://commons.wikimedia.org/wiki/File:Euclidean_vs_DTW.jpg>. Acesso em: 11 abr. 2013.

APÊNDICES

APÊNDICE A - QUESTIONÁRIO DE SATISFAÇÃO

Nome: _____ Idade: ____

1) Qual seu grau de familiaridade com o Sistema ITV?

- (a) Muito familiar
- (b) Familiar
- (c) Pouco familiar
- (d) Nenhum

2) Avalie o controle de interface do Sistema ITV através do Kinect.

- (a) Muito Satisfatório
- (b) Satisfatório
- (c) Pouco Satisfatório
- (d) Insatisfatório

3) Avalie a navegação no ambiente virtual do Sistema ITV através do Kinect.

- (a) Muito Satisfatório
- (b) Satisfatório
- (c) Pouco Satisfatório
- (d) Insatisfatório

4) Comparado ao mouse e ao teclado, a navegação no ambiente virtual através do Kinect é:

- (a) Muito Mais Fácil
- (b) Mais Fácil
- (c) Mais Difícil
- (d) Muito Mais Difícil

5) Avalie o reconhecimento de gestos no Sistema ITV.

- (a) Muito Satisfatório
- (b) Satisfatório
- (c) Pouco Satisfatório
- (d) Insatisfatório

6) Você teve mais dificuldade no reconhecimento de algum gesto específico? Qual?

- (a) Martelo
- (b) Chave de Fenda
- (c) Chave de Boca
- (d) Nenhum

7) Em sua opinião, quão interessante torna-se o aprendizado mediante a interação com Sistema ITV através do Kinect?

- (a) Muito Interessante
- (b) Interessante
- (c) Pouco Interessante
- (d) Desinteressante

Observações (Opcional):
