

UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

RENATO DE SOUSA GOMIDE

AMBIENTE VIRTUAL PARA REABILITAÇÃO DE MEMBROS
SUPERIORES UTILIZANDO VISÃO COMPUTACIONAL

Goiânia
2012

UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

RENATO DE SOUSA GOMIDE

**AMBIENTE VIRTUAL PARA REABILITAÇÃO DE MEMBROS
SUPERIORES UTILIZANDO VISÃO COMPUTACIONAL**

Dissertação apresentada ao Curso de Mestrado
em Engenharia de Computação da
Universidade Federal de Goiás, como parte
dos requisitos para obtenção do título de
Mestre em Engenharia de Computação.

Orientador: Prof. Dr. Marcus Fraga Vieira

Goiânia
2012

Dados Internacionais de Catalogação na Publicação (CIP)
GPT/BC/UFG/mr

633a Gomide, Renato de Sousa.
 Ambiente virtual para reabilitação de membros superiores
 utilizando visão computacional / Renato de Sousa Gomide. -
 2012.
 89 f. : il., figs.

 Orientador: Prof. Dr. Marcus Fraga Vieira.
 Dissertação (Mestrado) – Universidade Federal de Goiás,
 Escola de Engenharia Elétrica e de Computação, 2012.
 Bibliografia.
 Inclui lista de figuras, abreviaturas e siglas.

 1. Membros superiores – Reabilitação – Computador. 2.
 Ambiente virtual. 3. Visão computacional.

CDU:004.92:531/.534

“Ninguém baterá tão forte quanto a vida. Porém, não se trata de quão forte pode bater, se trata de quão forte pode ser atingido e continuar seguindo em frente. É assim que a vitória é conquistada.”

Silvester Stalone (Rocky Balboa – Rocky VI)

AGRADECIMENTOS

Aos meus pais, Eber e Dagmar, no qual me mostraram o valor de um trabalho honesto e por apoiar não apenas na formação acadêmica, mas também nos valores que formam o caráter de uma pessoa.

Aos meus avós maternos Aristides e Olímpia e a minha avó paterna Odete, que me fez perceber que o presente é sempre o melhor momento da vida e que não adianta guardar lamentações sobre um tempo que já passou.

Ao meu amigo e irmão mais velho Rodrigo que sempre me influenciou nas escolhas pessoais e profissionais.

A minha namorada Nathalya Rodrigues, que nesses últimos anos compartilhamos momentos de alegria e tristeza. Sou muito grato por tê-la ao meu lado, não há palavras que descreva a felicidade de estarmos juntos.

Aos meus amigos de infância, Frederico Mastrella, Luiz Loja, Marcelo Ioshida e Wadih Elkadi. Crescemos juntos e compartilhamos vários desafios e vitórias.

Aos colegas de graduação Ramon Campos e Marco Aurélio que também fizeram parte do Labioeng. Também aos colegas do Labioeng: Ivan Avelar, Fernanda Nora, Sara Andrade e Viviane Soares.

Ao terapeuta ocupacional Dagoberto Barbosa por ter disponibilizando seu tempo para conduzir as visitas técnicas realizadas no Centro de Reabilitação e Readaptação Dr. Henrique Santillo (CRER).

Aos professores da Universidade Federal de Goiás (UFG) Dr. Rodrigo Lemos e Dr. Thyago Marques por colaborarem com o ingresso no mestrado.

Ao professor Dr. Marcus Fraga Vieira pelo voto de confiança ao me aceitar como orientando. E também pelos momentos fora do ambiente de trabalho, mostrando o excelente ser humano que é.

Agradeço também a todas as outras pessoas que fizeram parte desta jornada de forma direta ou indireta, seja com uma simples conversa ou com ajudas essenciais para o andamento do projeto.

RESUMO

A utilização da computação em dispositivos de realidade virtual tem sido explorada para prover soluções na área da saúde, mais especificamente em recuperação funcional. Há estudos que indicam que a imersão causada pela utilização de ambientes virtuais em reabilitação apresenta resultados positivos sobre a evolução do quadro de um paciente em terapia motora. A partir desta informação, houve motivação para este estudo, que tem como objetivo principal desenvolver um ambiente virtual de baixo custo para recuperação funcional de membros superiores. O ambiente virtual foi modelado a partir de visitas técnicas realizadas em centro de reabilitação com o apoio de fisioterapeutas e terapeutas ocupacionais. Além disso, este trabalho descreve quais elementos internos do dispositivo computacional e também qual ambiente virtual foram desenvolvidos. O usuário interage com o ambiente virtual por meio de técnicas de visão computacional, tendo uma *webcam* como dispositivo de entrada de dados. As imagens adquiridas pela *webcam* são processadas de modo que um objeto de interesse fique em evidência por meio de segmentação da imagem. Foram analisadas cinco técnicas de segmentação de imagem nos espaços de cores RGB e HSV. Por meio dos resultados obtidos no ambiente de desenvolvimento, não foi possível classificar o melhor método de segmentação, pois o desempenho de cada um varia de acordo com a cor do objeto de interesse e da iluminação do ambiente externo. O ambiente virtual foi testado apenas em laboratório e em ambiente de desenvolvimento. Portanto, em parceria com profissionais da saúde, é necessário empregar o sistema em clínica de reabilitação para que haja a validação do ambiente virtual desenvolvido neste projeto.

Palavras-chave: Ambiente Virtual. Imagem Digital. Recuperação Funcional. Visão Computacional.

ABSTRACT

The use of computing devices used in virtual reality has been exploited to provide solutions in healthcare, specifically in functional rehabilitation. There are studies that indicate that immersion caused by the use of virtual environments in rehabilitation presents positive results on the evolution of a patient in therapy. From this information, there was motivation for this study, which aims at developing a low cost solution consisting in a virtual environment for upper limbs rehabilitation. This work describes the internal elements of the computing device and also the development of the virtual environment. The user interacts with the virtual environment through computer vision techniques, having a webcam as a data input device. Acquired images by the webcam are processed so that an object of interest may be in evidence by algorithms of image segmentation. There were analyzed five techniques of image segmentation in RGB and HSV color spaces. By the results obtained in the development environment, it wasn't possible to classify the best method of segmentation, and the performance of the methods varies according to the color of the object of interest and lighting of the external environment. The virtual environment was modeled after technical visits made in rehabilitation center supported by physiotherapists and occupational therapists. The virtual environment was tested only in research environment. Therefore it is necessary to employ the system in rehabilitation clinic in partnership with health professionals so that there is a validation of the virtual environment developed in this project in the area of rehabilitation.

Keywords: Computer Vision. Digital Image. Functional Rehabilitation. Virtual Environment.

LISTA DE ILUSTRAÇÕES

Figura 1 – (a) Microsoft™ Kinect (MICROSOFT, 2010); (b) PlayStation® Move (SONY, 2010); (c) Nintendo™ Wii Fit (NINTENDO, 2007).....	13
Figura 2 – Principais tipos de prejuízos nas funções motoras (adaptado de: CORPORAL (2011)).....	18
Figura 3 – Protótipo do Sensorama em uso (INVENTORVR, 2012)	21
Figura 4 – Sketchpad em operação (SUTHERLAND, 1964)	22
Figura 5 – Esquema de um dispositivo de entrada de dados baseado em luva virtual (ZIMMERMAN; LANNIER, 1991).....	23
Figura 6 – Exemplo de classificação dos elementos dos dispositivos de realidade virtual (adaptado de: GOUVEIA (1999))	24
Figura 7 – Processo de aquisição e visualização de uma imagem (adaptado de: GONZÁLEZ; WOODS (2008))	28
Figura 8 – Visualização dos pixels de uma imagem digitalizada.....	29
Figura 9 – Representação gráfica do espaço de cor RGB (MATHWORKS, 2011).....	31
Figura 10 – Representação geométrica do espaço de cor HSV (PIERCE, 2011)	33
Figura 11 – Comparação entre a distribuição espacial dos pixels de uma imagem nos espaços de cores RGB e HSV. (a) Imagem Original; (b) Pixels distribuídos no espaço de cor RGB; (c) Pixels distribuídos no espaço de cor HSV.....	34
Figura 12 – Imagem em diferentes escalas. (a) Imagem com 270 x 270 pixels; (b) Imagem com 40 x 40 pixels; (c) Imagem com 20 x 20 pixels.....	37
Figura 13 – Ajuste de brilho. (a) Imagem original; (b) Imagem com o brilho ajustado	38
Figura 14 – Ajuste de contraste; (a) Imagem original; (b) Imagem com baixo contraste; (c) Imagem com alto contraste.....	39
Figura 15 – Redução da paleta de cores. (a) 256 cores por canal; (b) 12 cores por canal.....	41
Figura 16 – Exemplo da ocorrência de <i>blobs</i> em uma imagem.....	42
Figura 17 – Operação de subtração entre o quadro atual e o quadro anterior	43
Figura 18 – Segmentação da imagem pelo método de subtração de background	43
Figura 19 – Histograma médio obtido no treinamento de cada componente RGB.....	44
Figura 20 – Segmentação efetuada pelo método de filtro por histograma de cores.....	45
Figura 21 – Coeficientes médios calculados no treinamento	45
Figura 22 – Segmentação efetuada pelo método de normalização de cores	46
Figura 23 – Distribuição espacial dos pixels da região de treino	47

Figura 24 – Segmentação utilizando indexação de tons. A região de treinamento é o retângulo branco na imagem à esquerda.....	48
Figura 25 – Representação espacial da região de treinamento do método de segmentação por tonalidade mínima e máxima.....	49
Figura 26 – Segmentação utilizando tonalidade mínima e máxima.....	49
Figura 27 – Iteração do algoritmo de dilatação	50
Figura 28 – Imagem após a execução do algoritmo de dilatação	50
Figura 29 – Resultado do algoritmo de dilatação aplicado à imagem segmentada da Figura 24; (a) Imagem original; (b) Imagem dilatada.....	51
Figura 30 – Fluxograma do processamento realizado pela <i>engine</i>	55
Figura 31 – Diagrama de classes da <i>engine</i>	57
Figura 32 – Diagrama de sequência do processo de trajetORIZAÇÃO de objetos de interesse.....	58
Figura 33 – Redução da escala da imagem. (a) Imagem original; (b) Escala de 1/2; (c) Escala de 1/4; Escala de 1/8.....	60
Figura 34 – Exemplo de redução da paleta de cores no espaço de cor RGB. (a) Imagem original; (b) 16 cores por canal RGB; (c) 4 cores por canal RGB; (d) 2 cores por canal RGB.....	61
Figura 35 – Demonstração de ajuste de brilho e contraste. (a) Imagem original; (b) Imagem ajustada	62
Figura 36 – Comparação entre métodos de segmentação utilizando um objeto de cor verde..	63
Figura 37 – Comparação entre métodos de segmentação utilizando um objeto de cor laranja durante o treinamento.	64
Figura 38 – Imagem segmentada com os <i>blobs</i> destacados	65
Figura 39 – <i>Engine</i> em execução.....	66
Figura 40 – Oclusão entre os membros. (a) Imagem original; (b) Imagem segmentada	67
Figura 41 – Painel de configuração da <i>engine</i>	68
Figura 42 – Esquema do exercício realizado em frente ao espelho.....	70
Figura 43 – Esboço do ambiente virtual.....	70
Figura 44 – Elementos gráficos do ambiente virtual. (a) Balão; (b) Balão quando é estourado; (c) Alfinete; (d) Plano de fundo.....	71
Figura 45 – Gráfico da trajetória realizada.....	73
Figura 46 – Trajetória realizada em relação à trajetória programada distribuídas no tempo ...	74
Figura 47 – Diagrama de classes do ambiente virtual.....	75
Figura 48 – Máquina de estados das classes herdeiras de <i>Interactive</i>	76
Figura 49 – Esquema da disposição dos objetos durante o uso do ambiente virtual.....	78

Figura 50 – Ambiente virtual na resolução de 1920 por 1080 pixels.....	78
--	----

LISTA DE ABREVIATURAS E SIGLAS

3D	Três Dimensões
AFE	Análise Funcional Escolar
AVE	Acidente Vascular Encefálico
BMP	<i>Windows Bitmap</i>
CMYK	<i>Cyan Magenta Yellow Key</i>
CI	Circuito Integrado
CAD	<i>Computer Aided-Design</i>
CRER	Centro de Reabilitação e Readaptação Dr. Henrique Santillo
DAM	Diário de Atividade Motora
DPI	<i>Dots Per Inch</i>
EQM	Erro Quadrático Médio
FAAST	<i>Flexible Action and Articulated Skeleton Toolkit</i>
GIF	<i>Graphics Interchange Format</i>
HSI	<i>Hue Saturation Intensity</i>
HSL	<i>Hue Saturation Luminance</i>
HSV	<i>Hue Saturation Value</i>
IHC	Interface Humano-Computador
JPEG	<i>Joint Photographic Experts Group</i>
OO	Orientação a Objetos
PNG	<i>Portable Network Graphics</i>
PPP	Pontos Por Polegada
RB2	Reality Built For Two
RGB	<i>Red Green Blue</i>
SNC	Sistema Nervoso Central
SUS	Sistema Único de Saúde
TA	Tecnologia Assistiva
UML	<i>Unified Modeling Language</i>
USAF	<i>United States Air Force</i>

SUMÁRIO

1	INTRODUÇÃO	13
2	OBJETIVOS	16
3	REVISÃO DE LITERATURA.....	17
3.1	Recuperação Funcional e Tecnologia Assistiva	17
3.1.1	<i>Neuroplasticidade</i>	<i>18</i>
3.1.2	<i>Tecnologia Assistiva.....</i>	<i>20</i>
3.2	Realidade Virtual	20
3.2.1	<i>Conceito de Realidade Virtual</i>	<i>23</i>
3.2.2	<i>Ambientes Virtuais Disponíveis</i>	<i>25</i>
3.3	Jogos Eletrônicos	26
4	IMAGEM DIGITAL	28
4.1	Composição da Imagem Digital	29
4.2	Espaço de Cor	30
4.2.1	<i>RGB</i>	<i>30</i>
4.2.2	<i>RGB Normalizado</i>	<i>31</i>
4.2.3	<i>HSV.....</i>	<i>32</i>
4.2.4	<i>Definição do Espaço de Cor Adequado</i>	<i>34</i>
5	PROCESSAMENTO DE IMAGENS DIGITAIS	36
5.1	Escala da Imagem.....	36
5.2	Ajuste de Brilho e Contraste	37
5.2.1	<i>Brilho.....</i>	<i>37</i>
5.2.2	<i>Contraste</i>	<i>38</i>
5.3	Redução da Paleta de Cores	40
5.4	Segmentação.....	41
5.4.1	<i>Subtração de Background</i>	<i>42</i>
5.4.2	<i>Filtro por Histograma de Cores.....</i>	<i>44</i>
5.4.3	<i>Normalização de Cores.....</i>	<i>45</i>
5.4.4	<i>Indexação de Tons.....</i>	<i>46</i>
5.4.5	<i>Tonalidade Mínima e Máxima</i>	<i>48</i>
5.5	Dilatação.....	49
6	MATERIAIS E MÉTODOS.....	52
7	ENGINE DO AMBIENTE VIRTUAL.....	54
7.1	Especificação.....	54

7.1.1	<i>Projeto</i>	56
7.1.2	<i>Aquisição das Imagens</i>	59
7.1.3	<i>Pré-Processamento</i>	59
7.1.4	<i>Segmentação</i>	62
7.1.5	<i>Análise da Imagem</i>	64
7.1.6	<i>Comunicação com Módulos Externos</i>	66
7.2	Resultados Obtidos	66
8	PROPOSTA DE AMBIENTE VIRTUAL	69
8.1	Modelo Conceitual	69
8.2	Informações de Saída	71
8.3	Arquitetura	75
8.4	Resultados e Discussões	77
9	CONCLUSÃO	80
9.1	Considerações Finais	81

1 INTRODUÇÃO

O mercado de jogos eletrônicos está em constante inovação juntamente com os consoles, que inovam em dispositivos de usabilidade alternativos. Os consoles da geração atual (Wii, PS3 e Xbox 360) possuem controladores diferentes dos *joysticks* tradicionais, os quais são compostos por um botão direcional e um conjunto de botões que expressam ações distintas. No mercado, há opções que trabalham com câmeras (Microsoft™ Kinect – Figura 1.a – e PlayStation® Move – Figura 1.b) ou sensores de posição e plataformas de força (Nintendo™ Wii Fit – Figura 1.c) (MAZALEK *et al.*, 2011).

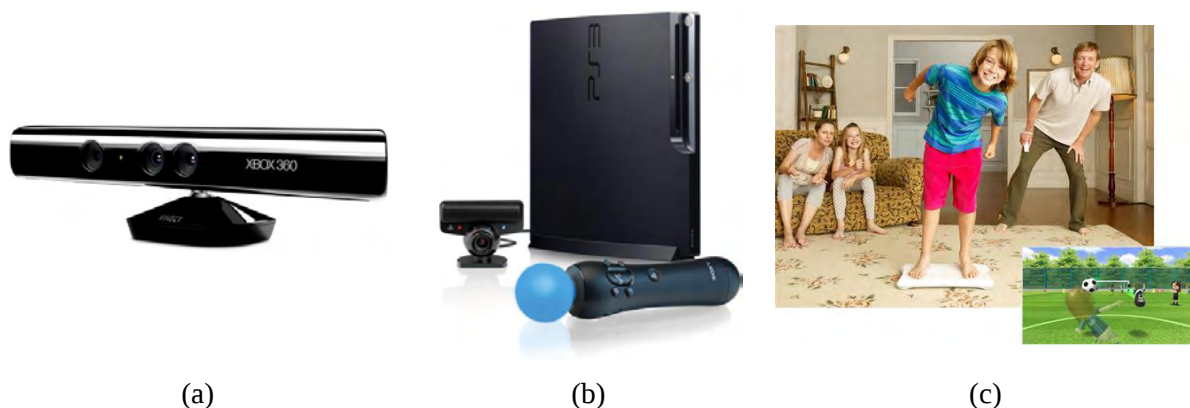


Figura 1 – (a) Microsoft™ Kinect (MICROSOFT, 2010); (b) PlayStation® Move (SONY, 2010); (c) Nintendo™ Wii Fit (NINTENDO, 2007)

Esta nova perspectiva abre campo não só para o avanço do entretenimento, mas também de jogos com outras propostas além da diversão do jogador. As áreas de alcance podem ir desde jogos pedagógicos até avançados simuladores utilizados em treinamentos militares. Este tipo de aplicação para jogos é conhecido como *serious game* (jogo sério), ou seja, é o jogo no qual o objetivo principal não é prover entretenimento para o jogador (COWLEY *et al.*, 2011).

Alguns jogos comuns são utilizados em clínicas de reabilitação com boa aceitação pelo fato de ser um estímulo extra à terapia clássica, permitindo que alguns pacientes consigam avançar no seu processo de recuperação (HALTON, 2008). Na literatura há estudos que favorecem a utilização deste recurso em terapia pelo fato de interagir com o usuário por meio de *feedback* audiovisual (ERTELT *et al.*, 2007). Porém, é importante deixar claro que este tipo de intervenção não tem a intenção de substituir o acompanhamento de um profissional da área de reabilitação.

Também há o fato de que os jogos utilizados para este fim possuem algumas restrições quanto à posição do jogador, que muitas vezes possui dificuldade em manter a postura exigida pelo jogo. Todavia, o grau de aceitação dos usuários à utilização de jogos

eletrônicos em terapia é alto. Por outro lado, do ponto de vista do terapeuta, alguns jogos não atendem às necessidades quanto à flexibilidade na sua configuração dos elementos contidos nos jogos, como por exemplo, dificuldade, física do mundo virtual, trilha sonora e efeitos visuais customizados (JOO *et al.*, 2010).

É importante para o terapeuta ajustar a dificuldade do jogo e a amplitude dos movimentos realizados durante a tarefa, pois cada paciente possui uma restrição motora diferente e também capacidade própria de manter-se concentrado durante a atividade. Logo, o terapeuta espera que vários perfis ou fases sejam elaborados conforme a necessidade de cada paciente.

Motivado pela perspectiva atual dos jogos eletrônicos, este projeto tem como objetivo desenvolver um ambiente virtual voltado para a recuperação funcional de membros superiores por meio de visão computacional. Uma das justificativas para este estudo está na carência de soluções de baixo custo e software livre disponíveis no mercado.

Além de propor uma solução de baixo custo, o dispositivo computacional desenvolvido neste trabalho poderá ser reaproveitado na criação de novos ambientes virtuais para que possam ser explorados outros membros do corpo humano. O mecanismo de interação, que é feito por meio de câmeras, também poderá ser utilizado em outras áreas além da de reabilitação, como por exemplo, reabilitação cognitiva, dispositivos de interação com o usuário, jogos eletrônicos entre outros.

Este trabalho está dividido em 9 Capítulos. O Capítulo de Introdução ou Capítulo 1 apresenta de forma geral o tema desenvolvido na dissertação, as justificativas para o estudo empreendido e a organização dada à dissertação. O Segundo Capítulo contém os objetivos gerais e específicos do projeto, descritos e analisados nesta dissertação.

O Capítulo 3 apresenta parte da bibliografia utilizada para a concepção e execução do projeto. Foram expostos de forma sintética os conceitos sobre reabilitação e tecnologia assistiva, que é a área fim do produto desenvolvido neste estudo. Também, com vistas a entender como procede a recuperação funcional sob o ponto de vista fisiológico, foi necessário levantar algumas informações sobre o fenômeno de neuroplasticidade e aplicações em tecnologia assistiva.

Os conceitos sobre realidade virtual e jogos eletrônicos também são expostos na revisão bibliográfica, uma vez que este trabalho consiste em desenvolver um ambiente virtual para fins de recuperação funcional de membros superiores. Além dos conceitos empregados neste tipo de projeto, também são citados alguns trabalhos que estão relacionados às áreas de pesquisa em tecnologia e saúde.

O ambiente virtual utiliza os recursos de visão computacional como mecanismo de interação entre o usuário e o sistema. O Capítulo 4 traz estudos sobre imagem digital, compreendida na formalização realizada quando uma imagem capturada do mundo real é representada no computador. Há vários tipos de representações para uma imagem digital. Este Capítulo mostra algumas representações que possuem boa aceitação entre os pesquisadores que tratam de visão computacional.

As imagens digitais podem ser alvo de uma série de ajustes e operações matemáticas para atingir algum objetivo, como por exemplo, realizar a equalização de brilho e cores ou até mesmo de realçar características de um elemento específico contido na imagem. Este tipo de operação é conhecido como processamento de imagens digitais. O Capítulo 5 tem o objetivo de explicar cada um dos tipos de operações de imagens digitais que foram utilizados no desenvolvimento do dispositivo computacional.

No Capítulo 6, trata-se de informações sobre os materiais e métodos utilizados para o desenvolvimento da pesquisa. A pesquisa foi dividida em duas etapas: a realização de visitas técnicas em centro de reabilitação com o intuito de modelar e codificar o ambiente virtual; e um período de desenvolvimento do dispositivo de interação entre usuário e computador.

Os Capítulos 7 e 8 descrevem as etapas realizadas no desenvolvimento do projeto sob duas perspectivas. Primeiramente, o Capítulo 7 enfatiza o desenvolvimento da *engine* do ambiente virtual, que é o mecanismo computacional responsável por fornecer a interação entre homem e computador. Já o Capítulo 8 mostra como o ambiente virtual proposto nesta pesquisa foi modelado e caracteriza as técnicas da codificação do ambiente virtual.

E, por fim, o Capítulo 9 expõe as conclusões alcançadas no estudo. As considerações finais também são relacionadas às perspectivas futuras de continuidade deste trabalho.

2 OBJETIVOS

Este trabalho tem como objetivo desenvolver um ambiente virtual em forma de jogo voltado para a recuperação funcional de membros superiores por meio de visão computacional. O processo de criação do ambiente virtual contempla a parte tecnológica para a criação do dispositivo de interação e também na modelagem da aplicação proposta.

O dispositivo desenvolvido tem como princípio manter o baixo custo de desenvolvimento e implantação, ou seja, o produto final é financeiramente acessível para os pacientes que estão realizando terapia de recuperação funcional de membros superiores.

O objetivo geral proposto para este trabalho pode ser dividido nos seguintes objetivos específicos:

- Estudar conceitos de realidade virtual, jogos eletrônicos e visão computacional para a criação do dispositivo de interação do ambiente virtual;
- Realizar a revisão de trabalhos sobre realidade virtual;
- Modelar o ambiente virtual com o auxílio de profissionais da área de fisioterapia e terapia ocupacional;
- Desenvolver o ambiente virtual e realizar os testes iniciais dentro do ambiente de pesquisa.

3 REVISÃO DE LITERATURA

Este capítulo aborda, de maneira geral, a revisão dos principais temas que fizeram parte do desenvolvimento deste trabalho. O trabalho pode ser dividido em uma parte tecnológica, que é o desenvolvimento do dispositivo computacional proposto, e uma parte relativa à aplicação, que é o emprego do dispositivo em recuperação funcional de membros superiores.

Serão abordados alguns conceitos sobre reabilitação e seu funcionamento sob o ponto de vista fisiológico. De certo modo, o processo de recuperação funcional varia para cada ser humano. Apesar de parecer subjetivo, há estudos que direcionam o processo de recuperação funcional para as terapias que apresentam melhores resultados.

Após a exposição dos conceitos sobre recuperação funcional, é apresentada uma visão histórica e conceitual sobre realidade virtual e jogos eletrônicos. Estes assuntos também foram contextualizados de acordo com o tema desta pesquisa. O emprego de ambientes virtuais em diversas áreas, especialmente na saúde, vem sendo explorado de forma bastante abundante no meio acadêmico.

3.1 Recuperação Funcional e Tecnologia Assistiva

O Sistema Nervoso Central (SNC) é responsável por processar as informações motoras, sensoriais e cognitivas de uma pessoa. Há diversos tipos de disfunções neurológicas ou lesões cerebrais que prejudicam tais funções. Este tipo de prejuízo pode afetar a qualidade de vida de uma pessoa e também pode se tornar um problema de saúde (FISHER; SULLIVAN, 2001).

Os prejuízos causados por lesões no sistema nervoso central e periférico variam de acordo com a região afetada. Os tipos de lesões com maior ocorrência causam danos nas funções motoras ou afetam as características cognitivas de uma pessoa (SCHALLERT; LEASURE; KOLB, 2000).

Os danos nas funções motoras, são classificados de acordo com a região motora afetada, podendo ser tetraplegia (membros inferiores e superiores), paraplegia (membros inferiores) ou hemiplegia (um dos lados do corpo afetados). A Figura 2 ilustra os tipos de danos nas funções motoras e as regiões do corpo afetadas.

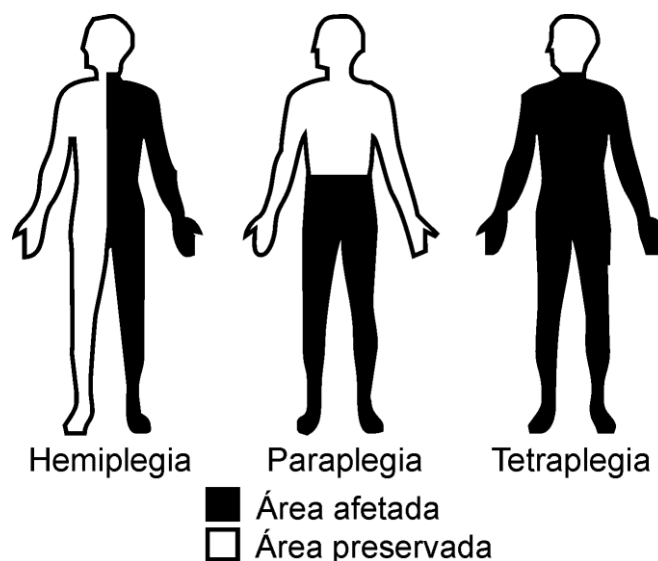


Figura 2 – Principais tipos de prejuízos nas funções motoras (adaptado de: CORPORAL (2011))

Uma função é caracterizada por comportamentos que cumprem o objetivo de uma tarefa em um ambiente relevante. A recuperação da função baseia-se na reconquista da função perdida após uma lesão. O objetivo principal da recuperação é permitir que uma pessoa lesionada possa cumprir o objetivo de uma tarefa que era capaz de efetuar antes da lesão (SHUMWAY-COOK; WOOLLACOTT, 2007).

Algumas vezes a recuperação é confundida com compensação. A compensação ocorre quando um indivíduo realiza a tarefa por meio de outra estratégia, ou seja, é uma substituição comportamental para a realização de um objetivo que era efetuado de maneira diferente.

A recuperação pode ser influenciada por alguns fatores, tais como: efeito da idade, característica da lesão, experiência sobre a atividade, efeito da farmacologia, treinamento e o acompanhamento clínico (HELD, 1998).

O processo de recuperação funcional tem como objetivo preliminar estimular a neuroplasticidade, que é a capacidade de adaptação do sistema nervoso às mudanças de condições do ambiente diário de uma pessoa (MINELLI, 2010).

3.1.1 Neuroplasticidade

Neuroplasticidade é definida pela capacidade de readaptação e organização do sistema nervoso às mudanças nas condições do ambiente diário de uma pessoa. Esta característica fisiológica está ligada aos processos de aprendizagem, memória e recuperação de lesões que afetam o sistema nervoso (LUNDY-EKMAN, 2008).

Durante a infância, a neuroplasticidade ocorre com maior frequência do que na fase adulta. A neuroplasticidade pode ocorrer por meio de modificações sinápticas, de modificações da membrana celular e por fatores neuroquímicos (ANDRADE; JUNIOR, 2005). Quando a neuroplasticidade é estimulada em indivíduos com lesões, nem sempre as funções perdidas são recuperadas. Em alguns casos, as funções podem ser mal adaptadas ao ponto de se tornarem funções patológicas (LENT, 2004).

O processo de recuperação neural deve ser direcionado para a função prejudicada. As estratégias de recuperação funcional consistem em estimular a reorganização neural por meio de treinamento repetitivo, sensorial, prática de tarefas cotidianas e prática mental. A evolução clínica de um paciente está ligada diretamente ao treino de atividades motoras prejudicadas (FISHER; SULLIVAN, 2001).

A experiência individual dentro de um tipo de atividade contribui para a recuperação. A plasticidade pode ser estimulada tanto por técnicas atividade-dependente ou aprendizagem-dependente.

A plasticidade atividade-dependente consiste em realizar a prática de atividades motoras como mecanismo de estímulo. A atividade-dependente é uma resposta ao treinamento que depende da demanda, repetição e dificuldade das tarefas exigidas (ADKINS *et al.*, 2006). Há indícios de que treinos de força isolados não estimulam a reorganização neuronal de forma significativa, e também que treinos com movimentos voluntários são mais eficientes que a realização de movimentos passivos auxiliados por dispositivos mecânicos (BYL *et al.*, 2003).

A plasticidade aprendizagem-dependente utiliza o treino de novas habilidades para promover a neuroplasticidade. A área de regiões ativas no encéfalo é maior quando um indivíduo está aprendendo uma nova habilidade do que quando já executa uma tarefa na qual possui experiência. Indivíduos com lesões corticais melhoram o desempenho motor nos 3 primeiros dias de treino de uma nova habilidade. Além disso, as alterações corticais foram fixadas de 7 a 10 dias após o treino (LANDERS, 2004).

São estudados novos recursos e serviços para que as habilidades funcionais de um indivíduo sejam ampliadas ou recuperadas. O produto gerado por este tipo de estudo é caracterizado como Tecnologia Assistiva (TA).

3.1.2 Tecnologia Assistiva

Tecnologia Assistiva é definida como “uma ampla gama de equipamentos, serviços, estratégias e práticas concebidas e aplicadas para minorar os problemas funcionais encontrados pelos indivíduos com deficiências” (COOK; HUSSEY, 2002).

Os equipamentos desenvolvidos em TA são organizados de acordo com os objetivos funcionais para o qual foram criados. Em 2002, houve a criação do padrão internacional ISO 9999/2002, intitulado como *Technical aids for persons with disabilities – Classification and terminology* que formaliza a organização, prescrição, estudo e pesquisa em TA (ISO, 2002).

O dispositivo proposto neste trabalho tem como finalidade auxiliar o paciente na terapia de recuperação funcional de membros superiores por meio de ambientes virtuais para computadores pessoais. Uma projeção futura para a pesquisa seria de explorar novos tipos de controladores para computadores utilizando dispositivos de entrada e saída de baixo custo.

3.2 Realidade Virtual

Desde a sua criação, os computadores eram utilizados em aplicações matemáticas com o objetivo de efetuar cálculos complexos. Com a expansão tecnológica dada pelo desenvolvimento dos transistores e circuito integrado (CI), os computadores diminuíram drasticamente de tamanho, permitindo que o uso de computadores no meio comercial fosse popularizado (TANENBAUM, 2006).

Os computadores começaram a ser utilizados com diversos propósitos, entre eles, aplicações que envolvem simulação e o aprendizado. Três pesquisadores, em períodos diferentes, desenvolveram projetos que direcionaram a perspectiva de usabilidade a um patamar elevado. Estes pesquisadores são Morton Heilig, Ivan Sutherland, Jason Lanier e Thomas Zimmermann, sendo que seus trabalhos são considerados ponto de partida em pesquisas relacionadas à realidade virtual.

Na década de 1960 foi desenvolvido por Morton Heilig o projeto chamado Sensorama. Este projeto consistia em uma cabine que tinha como finalidade prover um ambiente que despertava vários sentidos do ser humano por meio de imagens em três dimensões (3D), sistema de som estéreo, vibrações mecânicas, aromas e ar manipulado por ventiladores (HEILIG, 1998). O protótipo do Sensorama pode ser visto na Figura 3.



Figura 3 – Protótipo do Sensorama em uso (INVENTORVR, 2012)

O Sensorama reproduzia alguns filmes curtos, projetados especialmente para o dispositivo. Um dos filmes criados para o Sensorama era baseado em um passeio de bicicleta por um parque.

Os estudos e experimentos de Heilig contribuíram para a exploração de novas perspectivas de Interface Humano-Computador (IHC). Afirma-se que as primeiras aplicações que envolvem realidade virtual e cinema 3D foram desenvolvidas a partir dos protótipos de Heilig (HEILIG, 1998).

Em 1963, Ivan Sutherland apresentou o dispositivo Sketchpad em sua tese de PhD. O Sketchpad tinha como objetivo propor uma nova interface de interação com o computador por meio de uma ferramenta ótica. O usuário podia desenhar projetos diretamente na tela de um computador utilizando uma caneta ótica. Este tipo de solução deu origem às aplicações de computação gráfica do tipo Computer Aided-Design (CAD) (SUTHERLAND, 1964). Na Figura 4, o operador do sistema está segurando a caneta luminosa que interage com a tela do dispositivo.



Figura 4 – Sketchpad em operação (SUTHERLAND, 1964)

Sutherland também trabalhou em um projeto de um capacete para exibição de imagens, *The Ultimate Display*, o qual tinha como proposta prover uma janela para um mundo virtual. As imagens eram projetadas de acordo com a movimentação da cabeça do usuário, permitindo que o processo de criação dos trabalhos gráficos ficasse mais dinâmico e realista (SUTHERLAND, 1968).

No final da década de 70 foram explorados dispositivos computacionais de entrada alternativos baseados em luvas virtuais. Inicialmente desenvolvidas na Universidade de Illinois, as luvas virtuais transmitiam informações digitais sobre a posição da mão e articulação dos dedos por meio de transdutores (HAND, 1996).

Nos anos 80, os pesquisadores Thomas Zimmerman e Jason Lanier desenvolveram e patentearam dispositivos de realidade virtual que interagem com o usuário por meio de sensores ópticos, magnéticos e transdutores de articulação. A Figura 5 mostra o esquema de um dispositivo que posiciona o cursor no monitor por meio de sensores ópticos que processam a imagem de uma luva virtual com um marcador visual (ZIMMERMAN; LANNIER, 1991).

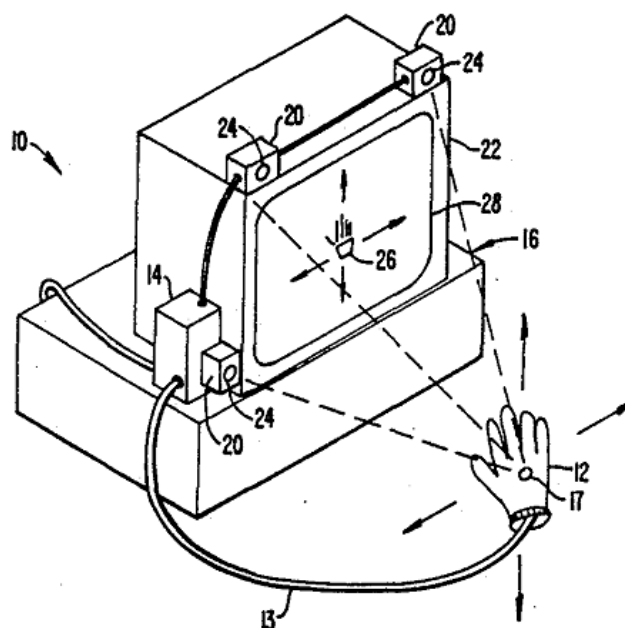


Figura 5 – Esquema de um dispositivo de entrada de dados baseado em luva virtual (ZIMMERMAN; LANNIER, 1991)

O grupo de pesquisa de Lanier (VPL Research Inc.) é conhecido por aprofundar os estudos em realidade virtual e o desenvolvimento de alguns produtos como: DataGlove (ZIMMERMAN *et al.*, 1986), EyePhone (FRANCHI, 1994), DataSuit (LASKO-HARVILL *et al.*, 1988) e Reality Built For Two (RB2) (BLANCHARD *et al.*, 1990).

A próxima subseção aborda a definição dos principais conceitos sobre realidade virtual. Estes conceitos são explorados durante o desenvolvimento de uma aplicação dessa natureza.

3.2.1 Conceito de Realidade Virtual

Realidade virtual é definida basicamente como um mundo interativo criado pelo computador no qual uma pessoa está com os seus sentidos voltados para a aplicação. Outra definição encontrada na literatura é de que a realidade virtual é uma maneira de visualizar, manipular e interagir com computadores por meio de dispositivos de entrada e saída alternativos. O mundo no qual uma pessoa faz as interações com os dispositivos de manipulação e visualização é chamado de ambiente virtual (KANADE;NARAYANAN; RANDER, 1995).

O ser humano possui a capacidade de usar a imaginação de forma abstrata, ou seja, o cérebro consegue trabalhar a percepção dentro de um processo cognitivo. A partir de imagens captadas, o cérebro constrói um mundo de forma convincente (WICKENS; BAKER, 1995).

A experiência da pessoa que utiliza um sistema de realidade virtual deve ser a mais próxima possível do mundo real. Porém, existem limitações tecnológicas e financeiras que não permitem a realização desta experiência com total precisão. Esta precisão é chamada de imersão, que corresponde à concentração da pessoa no mundo virtual e retira a possibilidade de uso dos sentidos na percepção do mundo físico (PAUSCH;PROFFITT; WILLIAMS, 1997).

Uma aplicação de realidade virtual pode ser desenvolvida também em computadores pessoais utilizando dispositivos convencionais. Este tipo de aplicação é conhecido como sistema de realidade virtual *desktop*. O usuário utiliza o próprio monitor como dispositivo de visualização e o teclado, mouse ou *webcams* como dispositivos de manipulação. O grau de imersão nos ambientes *desktops* é menor comparado aos sistemas mais avançados (ROBERTSON;CZERWINSKI; DANTZICH, 1997).

Do ponto de vista técnico, quando se trabalha com projetos de realidade virtual, três elementos devem ser explorados: monitoração, controle e visualização (GOUVEIA, 1999). O aspecto de monitoração diz respeito a como os dispositivos de entrada farão a captura da intenção do usuário em relação à entrada de dados. A parte de controle refere-se ao comportamento dos objetos do mundo virtual. E a visualização é representada pelos aspectos audiovisuais do ambiente e, em alguns casos, dispositivos táteis. A Figura 6 exemplifica a classificação dos dispositivos nestas três áreas de estudo.

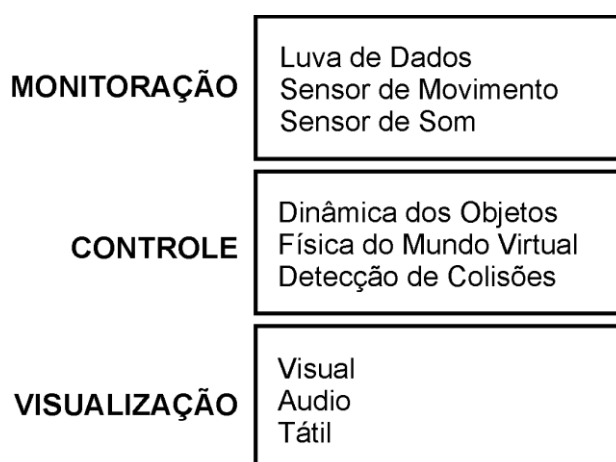


Figura 6 – Exemplo de classificação dos elementos dos dispositivos de realidade virtual (adaptado de: GOUVEIA (1999))

A aplicação de ambientes virtuais está presente em diversas áreas de interesse, como entretenimento, aprendizagem, desenvolvimento de produtos, medicina, recuperação funcional, arquitetura, simuladores de voo entre outros.

Foram pesquisados vários ambientes virtuais documentados na literatura que tem o propósito similar ao deste trabalho. Para realizar a busca foram utilizadas as bases de dados PubMed e Google Academics. As palavras chaves foram *virtual environment*, *microsoft kinect* e *computer vision tracking*. As buscas foram realizadas para os trabalhos publicados a partir do ano 2000.

3.2.2 Ambientes Virtuais Disponíveis

Durante a pesquisa, foram encontrados alguns trabalhos que exploram ambientes virtuais. Esses trabalhos, em alguma medida, influenciaram o desenvolvimento deste projeto. Alguns pelo fato de trabalharem a proposta de monitoração por meio de dispositivos semelhantes e outros porque tinham o tipo de aplicação voltado para a recuperação funcional.

A pesquisa teve como ponto de partida o trabalho de revisão voltado para os tipos de sensores de movimento (ZHOU; HU, 2008). Na revisão, os referidos autores especificam os vários tipos de sensores de movimento que são utilizados em pesquisas que envolvem o estudo do movimento humano. Os estudiosos classificaram os sensores por categorias: não-visuais, visuais e mistos.

Os sensores não-visuais são aqueles que são fixados no corpo humano com o objetivo de coletar informação da movimentação. Estes sensores na maioria das vezes realizam a comunicação por sinais de natureza mecânica, de inércia, acústico, por rádio, magnético ou por micro-ondas.

A utilização de câmeras está relacionada aos sensores visuais. Os sensores visuais podem ser agrupados de acordo com o uso de marcadores visuais fixados ao objeto de estudo. O uso de marcadores visuais dependerá do tipo de aplicação que será realizada.

Os sensores podem também ser agrupados de forma mista para que ocorra a extração dos principais benefícios de cada tipo de sistema com o intuito de minimizar o erro na coleta dos dados.

O trabalho de MAURI *et al.* (2006) propõe um ponto de partida para pesquisas aplicadas em IHC. O objetivo principal do trabalho consiste em desenvolver técnicas que auxiliam pessoas com paralisia cerebral a controlarem o computador por meio de visão computacional. A monitoração é realizada após a identificação de marcadores fluorescentes fixados na pessoa. A aplicação desenvolvida disponibiliza para o usuário o controle do mouse pelo rastreamento do marcador e um teclado virtual.

SUMA *et al.* (2011) desenvolveram um *framework* chamado Flexible Action and Articulated Skeleton Toolkit (FAAST) para trabalhar com o sensor de movimento Kinect

(BLEIWEISS *et al.*, 2010). Este *framework* possibilita que a linguagem corporal do usuário seja traduzida em comandos para aplicações e jogos de computadores pessoais.

BOTELLA *et al.* (2011) criaram um ambiente virtual para o tratamento de pessoas com fobia a baratas. Foi desenvolvido um jogo sério para dispositivos móveis e computadores pessoais no qual a imersão consiste na projeção de baratas que interagem com o usuário. Utilizando a mesma linha de raciocínio, o trabalho desenvolvido por ASSIS *et al.* (2008) utilizou os recursos de realidade aumentada para tratamento de dor fantasma em pacientes com o braço amputado.

Alguns projetos trabalham com mais de um tipo de sensor para realizar a monitoração e o controle do ambiente. No estudo de NACKE *et al.* (2011), a equipe desenvolveu um controlador para jogos eletrônicos baseado em *biofeedback* misturando a resposta de vários sensores fisiológicos. O sistema é flexível quanto à sua configuração. Por exemplo, parte do controle pode ser feito pelo movimento dos olhos enquanto outros comandos são realizados por eletromiografia e termômetros.

Existem diversos trabalhos que envolvem a utilização de realidade virtual em computadores pessoais. Nesta subseção, foram apresentados apenas alguns deles que tiveram importância na definição e na execução da pesquisa realizada neste projeto.

O sistema desenvolvido constitui-se de um mecanismo que permite a interação entre usuário e computador utilizando um ambiente virtual. Esta aplicação baseia-se em jogos eletrônicos voltados para recuperação funcional de membros superiores. A próxima subseção descreve os elementos básicos que um jogo eletrônico possui.

3.3 Jogos Eletrônicos

Jogos eletrônicos são basicamente programas de computadores que na maioria das vezes tem como objetivo promover o entretenimento do jogador. Diferente dos softwares convencionais, os jogos eletrônicos demandam constante atualização dos dados e da tela onde está sendo projetado (OVERMARS, 2004).

Há um grau de interatividade maior para os jogadores em relação aos softwares convencionais, nos quais os jogadores devem permanecer concentrados na tarefa realizada. Todos os jogos realizam a criação de um mundo virtual no qual o jogador encontra-se imerso, mundos virtuais que podem ou não simular o mundo real (FUNK, 1993).

Como a quantidade e a variedade de jogos disponíveis no mercado é alta, surgiram alguns critérios que permitem a classificação dos jogos eletrônicos. Os jogos podem ser classificados dentro dos seguintes critérios: dimensão (a dimensão espacial do mundo

virtual criado), ponto de vista (como o personagem do jogador é apresentado na tela), gênero (características de jogabilidade e intenção de cada jogo) e número de jogadores (podendo variar de um até centenas de jogadores).

A dimensão de um jogo corresponde à dimensão espacial do mundo virtual criado, pode ser tanto em duas ou três dimensões. O ponto de vista corresponde em como o personagem do jogador é apresentado na tela. O personagem pode ser controlado sobre uma perspectiva de primeira pessoa, quando o jogo simula a inserção do jogador dentro do mundo virtual, ou em terceira pessoa, em que geralmente o jogador vê o personagem sob uma visão superior.

O gênero dos jogos varia de acordo com as características de jogabilidade e intenção de cada jogo. Os jogos são classificados em diversos gêneros, como por exemplo, jogos de esporte, simuladores de corrida, ação entre outros. O número de jogadores simultâneos dentro do mesmo mundo virtual pode variar de um jogador (*singleplayer*) e em até centenas de jogadores (*multiplayer*) ao mesmo tempo (KAMEL).

4 IMAGEM DIGITAL

A imagem digital pode ser definida como a representação binária de uma imagem bidimensional adquirida do mundo real ou produzida artificialmente. Uma imagem digital deve ser capaz de interagir com ações de aquisição, visualização, armazenamento e processamento (KOSCHAN; ABIDI, 2008).

Aquisição pode ser entendido como o mecanismo que codifica uma imagem de natureza contínua para o formato digital. Alguns dispositivos que fazem parte do cotidiano são os *scanners*, câmeras digitais e filmadoras digitais. A visualização de uma imagem digitalizada pode ser reconstruída em superfícies como monitores, televisores e impressoras.

A Figura 7 ilustra como é realizado o processo de aquisição e visualização de uma imagem.

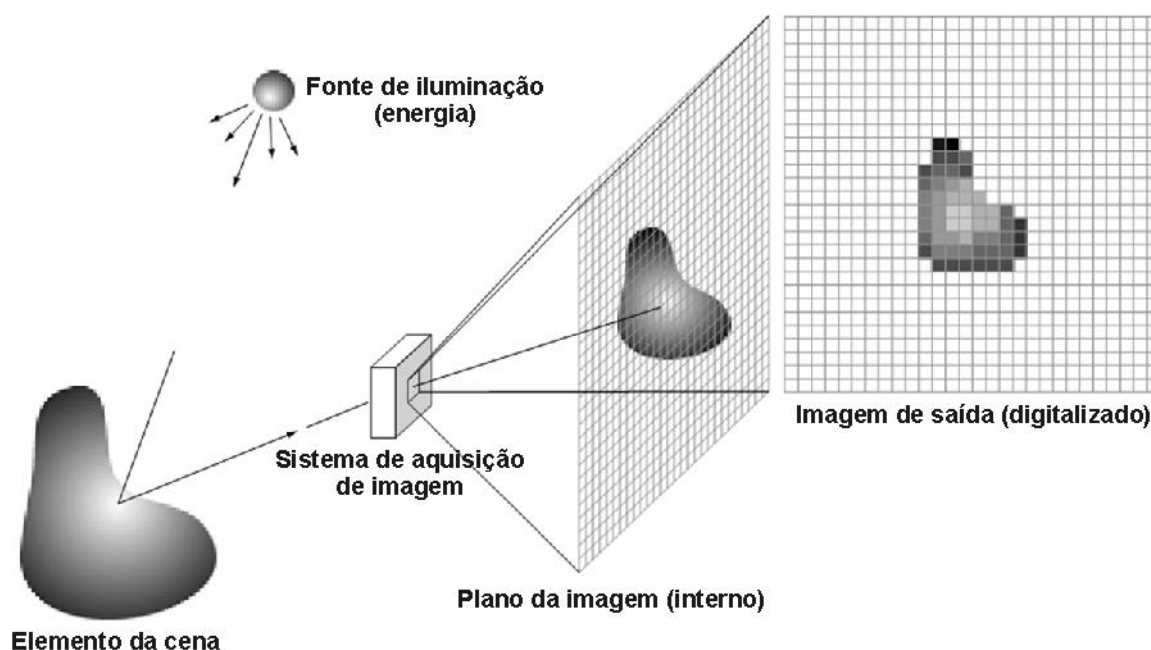


Figura 7 – Processo de aquisição e visualização de uma imagem (adaptado de: GONZÁLEZ; WOODS (2008))

Existem vários padrões de armazenamento de uma imagem digital como, por exemplo, *Joint Photographic Experts Group* (JPEG), *Graphics Interchange Format* (GIF), *Windows Bitmap* (BMP) e *Portable Network Graphics* (PNG). Cada um destes padrões possuem sua forma de codificação e particularidades para tipos de aplicação específicos (STONE, 2003).

Já o processamento da imagem é definido como operações computacionais realizadas sobre uma imagem digital para alcançar um objetivo específico. Estes objetivos variam desde retoques na imagem para o uso em mídias impressas a aplicações complexas de

sensoriamento remoto, que exigem a extração de informações de imagens adquiridas por satélites (GONZÁLEZ; WOODS, 2008).

As próximas subseções apresentam os conceitos básicos para o entendimento do padrão de imagem utilizado nesta pesquisa. Também serão descritos quais foram os tipos de processamento utilizados na aquisição das informações necessárias para o funcionamento da *engine* proposta.

4.1 Composição da Imagem Digital

De modo geral, uma imagem digital é representada computacionalmente por uma matriz algébrica de números inteiros na qual cada ponto da matriz expressa a informação de uma cor. Este ponto da matriz é denominado *picture element* ou pixel, ou seja, um pixel é a menor componente da imagem digital (KOSCHAN; ABIDI, 2008).

A dimensão de uma imagem é dada pela quantidade de pixels. Esta unidade de medida não tem relação com o sistema métrico convencional. A relação entre a quantidade de pixels da imagem em relação ao tamanho equivalente pelo sistema métrico é definida pela resolução. A unidade de medida da resolução de uma imagem é definida por Pontos Por Polegada (PPP) ou *Dots Per Inch* (DPI).

Quanto maior a resolução da imagem, maior é o nível de ampliação que esta pode sofrer sem que ocorram distorções que prejudiquem a qualidade visual no momento de visualização.

A imagem da Figura 8 possui a resolução de 389 x 389 DPI. Uma área da imagem foi ampliada de modo que seja possível visualizar a matriz de pixels do trecho ampliado.



Figura 8 – Visualização dos pixels de uma imagem digitalizada

O modelo que representa a cor de cada pixel é chamado de espaço de cor. O espaço de cor é um modelo matemático responsável por abstrair a constituição física da cor e formalizá-la conforme uma metodologia definida. Os espaços de cores mais utilizados são:

Tons de Cinza, *Red Green Blue* (RGB), *Hue Saturation Value* (HSV), *Cyan Magenta Yellow Key* (CMYK) e Preto e Branco.

4.2 Espaço de Cor

Uma cor pode ser representada matematicamente por meio de um espaço de cor. Cada tipo de espaço de cor possui uma abordagem diferenciada. As cores visíveis para os olhos humanos podem ser expressas de diferentes formas.

Nos experimentos realizados neste projeto foram utilizados os espaços de cores RGB, HSV e Preto e Branco. Os modelos RGB e HSV foram estudados com o objetivo de saber qual espaço de cor fornece melhor resultado no processo de segmentação de objetos na imagem.

A segmentação da imagem consiste em separar o que é considerado primeiro plano (*foreground*) do plano de fundo (*background*). Posteriormente, serão apresentadas as técnicas utilizadas para realizar a segmentação da imagem.

4.2.1 RGB

O espaço de cor RGB consiste na mistura das cores vermelha (R), verde (G) e azul (B). O modelo RGB é um dos modelos de representação de cor mais simples e intuitivo (BRAND; MASON, 2000). Graficamente, esse espaço de cor é representado por um sistema de coordenadas em um plano tridimensional, no qual cada eixo corresponde às componentes de cores primárias.

A representação espacial do espaço de cor RGB é ilustrada na Figura 9. Neste exemplo, os valores das componentes são números inteiros que vão de 0 a 255. Cada pixel de uma imagem com esta configuração de cor possui 24 bits, sendo 8 bits para representar cada componente de cor.

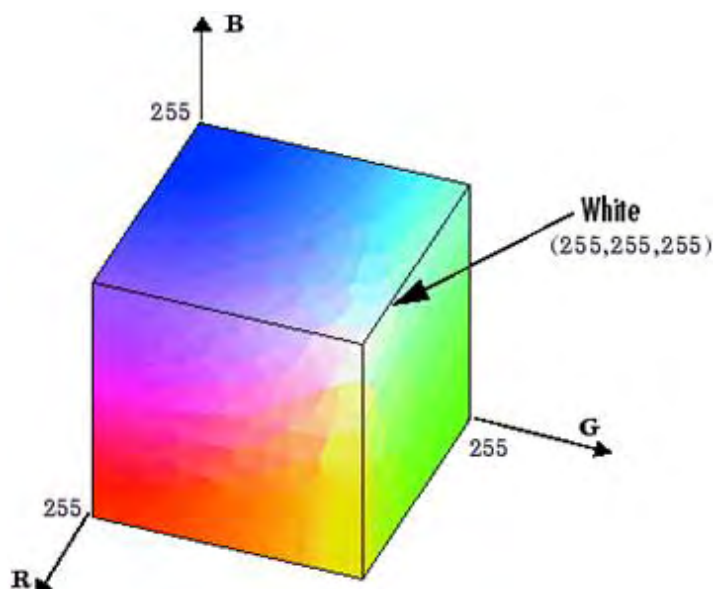


Figura 9 – Representação gráfica do espaço de cor RGB (MATHWORKS, 2011)

A mistura das componentes de cores dá origem às cores secundárias, sendo que cada cor pode ser expressa por R, G e B. Uma característica relevante deste modelo é que a distância entre duas cores é obtida pela distância Euclidiana (HECKBERT, 1982).

Do ponto de vista prático, uma cor pode ser associada facilmente com suas cores vizinhas, principalmente quando esta comparação é realizada a olho nu. Quando a cor é analisada por meio de métodos computacionais, cores vizinhas ou indesejadas podem fazer parte de uma amostra específica.

Por exemplo, quando o processamento da imagem é feito em imagens obtidas por vários frames de um vídeo e utiliza as cores como variáveis de ambiente da análise, pequenas variações de iluminação nas imagens podem prejudicar o experimento realizado.

4.2.2 RGB Normalizado

Visando contornar a fragilidade do espaço de cor RGB perante as variações de iluminação na imagem, foi desenvolvido o modelo matemático de normalização. Este método consiste em calcular a intensidade total de uma cor pelo somatório da intensidade das componentes RGB e obter os componentes normalizados por meio da divisão de cada componente pela intensidade total.

As equações de (1) a (4) formalizam o processo de normalização de uma cor no espaço RGB.

$$I = R + G + B \quad (1)$$

onde I é a intensidade expressa pela soma da intensidade das cores R , G e B .

$$r = R/I \quad (2)$$

$$g = G/I \quad (3)$$

$$b = B/I \quad (4)$$

$$r + g + b = 1 \quad (5)$$

Nota-se pela equação (5) que o somatório das componentes normalizadas r , g e b é sempre igual a um. Por meio desta relação, a componente b pode ser omitida e facilmente recuperada, podendo haver a redução da dimensão espacial na representação normalizada.

Os componentes r e g são considerados como as cores puras deste modelo e a componente b como a intensidade do brilho. Ao ignorar a luz de uma superfície opaca, o RGB normalizado torna-se invariante às mudanças de orientação das fontes de luz na superfície (WANG; SUNG, 1999). Esta característica permite que a robustez seja maior em aplicações que apresentam variações na iluminação do ambiente.

Há espaços de cores que trabalham com a ideia de tonalidade, saturação e brilho. Esta abordagem difere-se da RGB, pois a cor não é considerada uma mistura de componentes primárias. Neste trabalho, foi estudado o espaço de cor HSV, que é uma das abordagens que possui a característica de formalizar a cor por meio de tonalidade.

4.2.3 HSV

O espaço de cor HSV representa a cor por variáveis de tonalidade, saturação e valor. A tonalidade corresponde à cor pura, enquanto a intensidade da cor é descrita pela saturação. O aspecto do valor é caracterizado pela diferença entre o claro e escuro de uma cor.

A representação gráfica do espaço de cor HSV é ilustrada pela forma geométrica de um cone (fig. 10). De acordo com a Figura 10, para definir uma cor, primeiramente, é necessário definir o ângulo H , que representa a tonalidade. Logo após, é escolhido a saturação que na figura corresponde ao raio S . Por fim a altura define a iluminação pela variável V .

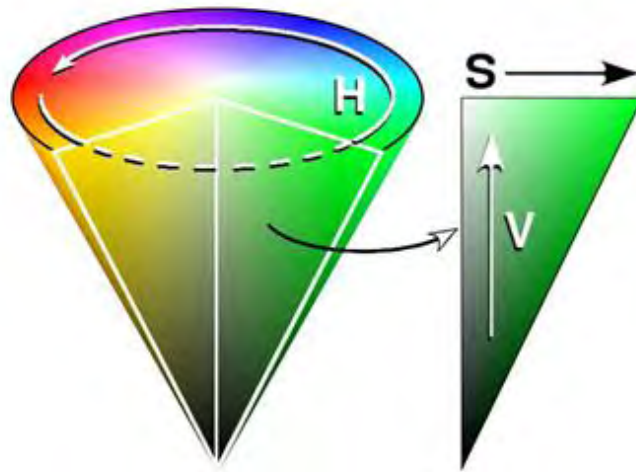


Figura 10 – Representação geométrica do espaço de cor HSV (PIERCE, 2011)

Este modelo de representação das cores possui a vantagem de separar as componentes de iluminação e tonalidade da cor. Diferente do modelo RGB, cores em mesmas condições de iluminação diferem-se pelo parâmetro de tonalidade, permitindo uma análise mais robusta em alguns casos (SAXE; FOULDS, 1996).

Algumas linguagens de programação como *Action Script 3*, *Matlab*, *Java* e *C#*, possuem funções de leitura de imagens no espaço de cor RGB. Muitas vezes, para trabalhar com a cor no espaço HSV, necessita-se implementar algoritmos que convertem uma cor de RGB para HSV.

A conversão de uma cor do modelo RGB para HSV não é linear. Todavia, o cálculo da tonalidade possui algumas operações lógicas não muito complexas. Segundo (TRAVIS, 1991), as equações de (6) a (8) formalizam esta conversão.

$$H = \begin{cases} 60 * \frac{G - B}{MAX - MIN} + 0, se MAX = R e G \geq B \\ 60 * \frac{G - B}{MAX - MIN} + 360, se MAX = R e G < B \\ 60 * \frac{B - R}{MAX - MIN} + 120, se MAX = G \\ 60 * \frac{R - G}{MAX - MIN} + 240, se MAX = B \end{cases} \quad (6)$$

$$S = \frac{MAX - MIN}{MAX} \quad (7)$$

$$V = MAX \quad (8)$$

onde MAX e MIN correspondem respectivamente ao maior e menor valor de intensidade das componentes RGB. O valor de H é definido em graus (°), enquanto os valores de S e V estão entre 0 e 1.

Há outros espaços de cores que trabalham com a tonalidade e elementos que atuam sobre o tom de uma cor, como por exemplo, os espaços de cores *Hue Saturation Luminance* - HSL e *Hue Saturation Intensity* - HSI.

4.2.4 Definição do Espaço de Cor Adequado

Soluções que utilizam processamento de imagens e vídeo devem ser projetados para utilizar espaços de cores específicos. O pesquisador ou desenvolvedor deve escolher os espaços que são adequados ao problema proposto. Não há um espaço de cor padrão para o uso em pesquisas acadêmicas. Há trabalhos que utilizam espaços de cores diversos para a solução de problemas parecidos, como podem ser vistos em VEZHNEVETS;SAZONOV; ANDREEVA (2003).

A Figura 11 ilustra um exemplo da distribuição dos pixels de uma mesma imagem nos espaços de cores RGB e HSV. Nota-se que a composição das cores de cada esfera presente na imagem é bem definida em ambos os espaços de cores. Porém, cada elemento não possui a cor pura no espaço de cor RGB. Na composição da esfera vermelha, por exemplo, os pixels possuem componentes azul e verde. Já no espaço de cor HSV, as esferas são separadas pela característica da tonalidade. Esta característica pode ser considerada como decisivo na escolha da estratégia adotada em algoritmos de segmentação de imagem.

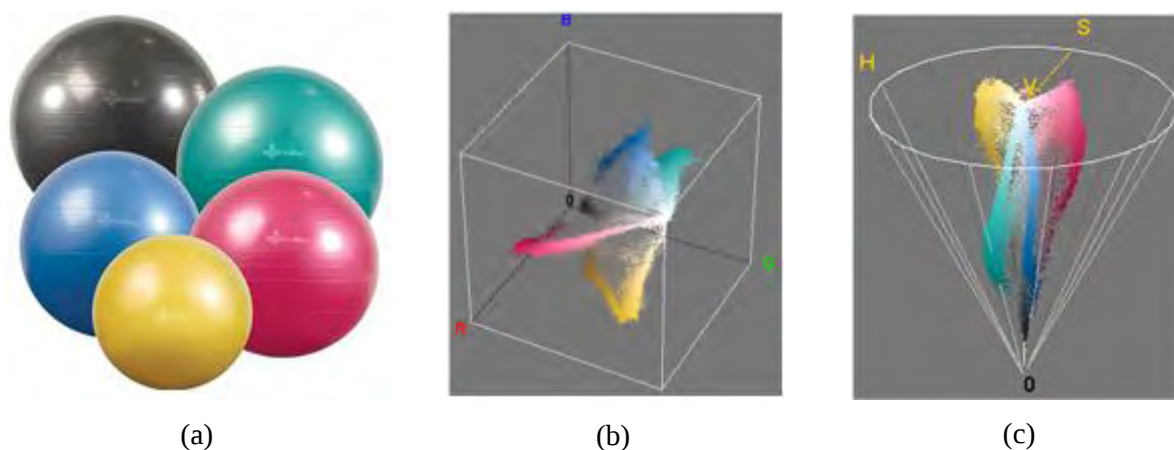


Figura 11 – Comparação entre a distribuição espacial dos pixels de uma imagem nos espaços de cores RGB e HSV. (a) Imagem Original; (b) Pixels distribuídos no espaço de cor RGB; (c) Pixels distribuídos no espaço de cor HSV.

No que diz respeito ao problema proposto – detectar e rastrear um objeto pela cor – existem resultados aceitáveis utilizando técnicas distintas com espaços de cores distintos. Alguns trabalhos propõem a comparação dos resultados obtidos utilizando espaços de cores diferentes no mesmo experimento (ZARIT;SUPER; QUEK, 1999).

Há de se considerar, ainda, que os resultados nos métodos de segmentação da imagem não produzem informações conclusivas sobre a influência dos espaços de cores adotados. Neste trabalho foram utilizados os modelos RGB, RGB Normalizado e HSV. A descrição e análise dos resultados serão detalhadas na descrição da *engine* proposta. Pode-se antecipar que os melhores resultados foram obtidos com os espaços de cores RGB Normalizado e HSV.

5 PROCESSAMENTO DE IMAGENS DIGITAIS

Este capítulo aborda os métodos de processamento de imagens que foram utilizados nesse trabalho. O processamento da imagem pode ser realizado tanto no domínio do tempo quanto no da frequência. As técnicas citadas foram todas realizadas no domínio do tempo no espaço de cor RGB.

Primeiramente, serão mostradas as operações utilizadas como pré-processamento da imagem digital, tais como ajuste de brilho e contraste, redução da escala e redução da paleta de cores. Como pós-processamento serão descritas as operações de segmentação e de dilatação.

5.1 Escala da Imagem

Uma imagem após a aquisição é armazenada em duas dimensões (largura e altura) mensurada em pixels. Pode ser que para uma determinada aplicação, a dimensão da imagem processada seja satisfatória. Contudo, em alguns casos, há a necessidade de reduzir ou aumentar a dimensão da imagem. No processamento de imagens, a alteração da escala é um dos primeiros ajustes a ser realizado, tendo em vista que alguns algoritmos efetuam a leitura de todos os pixels da imagem. Logo, pode haver o ganho de desempenho ao diminuir a escala da imagem.

Entende-se como escala um fator numérico que representa a proporção no qual a imagem é ampliada ou reduzida. Ou seja, se uma operação de redimensionamento for realizada com a escala de $\frac{1}{4}$, quer dizer que a imagem é reduzida em quatro vezes em relação ao seu tamanho original.

A maioria das linguagens de programação e *toolboxes* de processamento de imagens apresenta métodos para alteração da escala e redimensionamento da imagem. Do ponto de vista do desenvolvedor, não há controle sobre o algoritmo de redimensionamento, ou seja, o funcionamento da operação de escala é transparente para o usuário. Na execução do método de redimensionamento, basta informar a nova escala como parâmetro de entrada.

Os algoritmos para alteração de escala da imagem mais utilizados são: *nearest-neighbor interpolation* (LEHMANN;GONNER; SPITZER, 1999), *bilinear interpolation* (GRIBBON; BAILEY, 2004) e *bicubic interpolation* (KEYS, 1981). Cada um desses algoritmos possui uma estratégia diferente na hora de calcular a imagem resultante nas operações de redução ou ampliação.

A Figura 12 apresenta uma mesma imagem dimensionada em três formas distintas. A primeira imagem está com 270 x 270 pixels, a segunda com 40 x 40 pixels e a

terceira com 20 x 20 pixels. Nota-se que a primeira imagem está com uma definição aceitável para este nível de ampliação (sem apresentar serrilhado), enquanto as demais estão distorcidas.

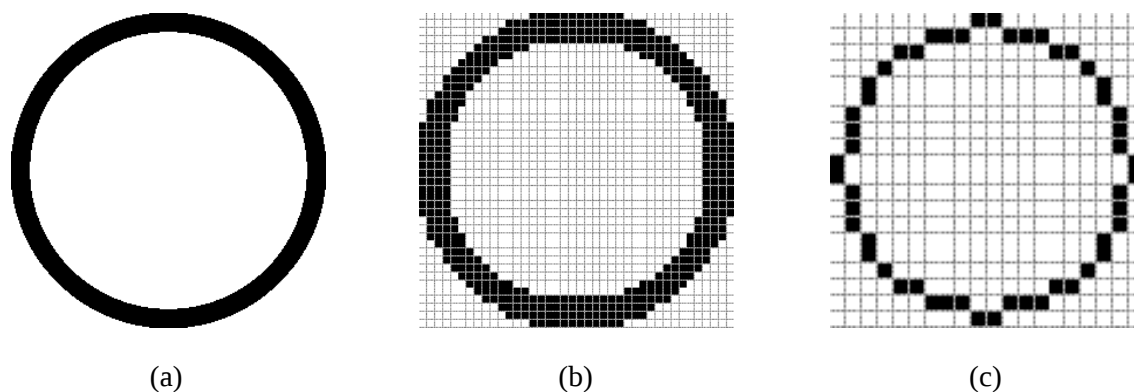


Figura 12 – Imagem em diferentes escalas. (a) Imagem com 270 x 270 pixels; (b) Imagem com 40 x 40 pixels; (c) Imagem com 20 x 20 pixels

O processo de alteração da escala da imagem no pré-processamento possibilita a melhoria no desempenho de todo o processamento da imagem, pois alguns algoritmos utilizados no projeto realizam a leitura pixel a pixel da imagem. Porém, deve haver a precaução de estabelecer uma escala para que a imagem não fique muito pequena ao ponto de prejudicar o restante da análise desta.

5.2 Ajuste de Brilho e Contraste

Os ajustes de brilho e contraste permitem ajustar na imagem a intensidade entre as cores claras e escuras e também se uma imagem é mais opaca ou brilhante. Cada um desses ajustes é efetuado deslocando o histograma da imagem para as operações de brilho, ou esticando ou encolhendo o histograma nas operações de contraste (GONZÁLEZ; WOODS, 2008).

5.2.1 Brilho

O brilho de uma imagem está ligado à iluminação da imagem de um modo geral. Se uma imagem encontra-se escura, é necessário que o brilho seja aumentado para que a percepção visual da imagem seja natural.

Para realizar o ajuste de brilho em uma imagem basta somar um valor escalar para cada pixel da imagem. Sob o ponto de vista do histograma da imagem, a distribuição dos pixels é deslocada de acordo com o valor escalar escolhido. Se o valor escalar é positivo, o histograma desloca-se para a direita, caso contrário para a esquerda.

A Figura 13 mostra uma imagem e seu histograma antes e depois do ajuste de brilho. Nota-se que, como o ajuste realizado aumentou a iluminação da imagem, o histograma foi deslocado para a direita.

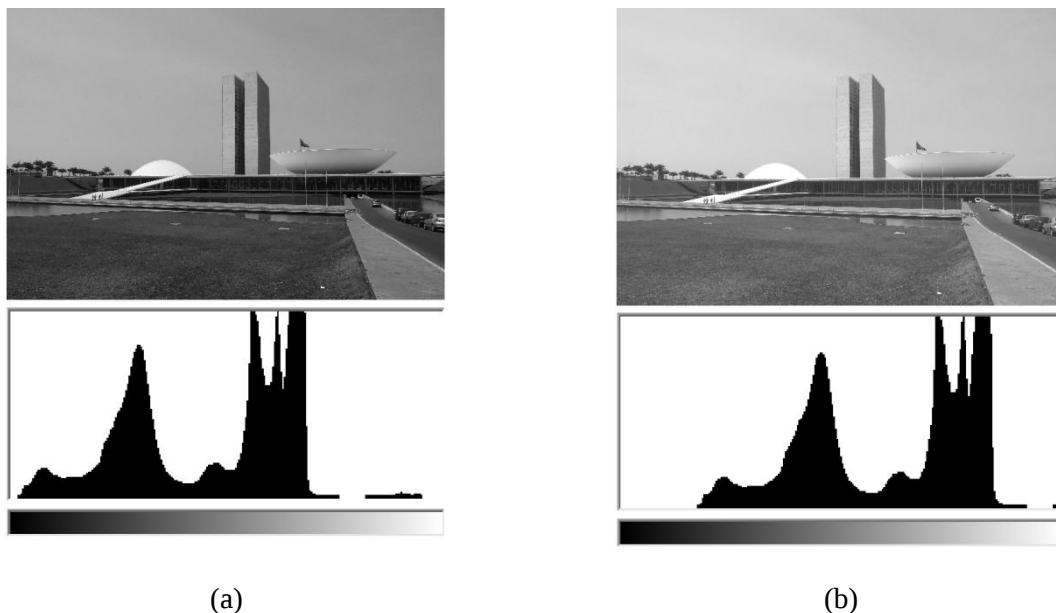


Figura 13 – Ajuste de brilho. (a) Imagem original; (b) Imagem com o brilho ajustado

Após o ajuste, alguns pixels extrapolaram o valor do limite superior. Como o espaço de cor possui seus valores máximos e mínimos estabelecidos, estes pixels extrapolados são ajustados para os valores limites definidos pelo espaço de cor, que neste caso é 0 ou 255. No exemplo apresentado, os pixels extrapolados representam a cor branca (que neste caso é o valor máximo no espaço de cor RGB).

5.2.2 Contraste

Diferente do brilho, o contraste estabelece a representação de quanto uma cor é mais clara ou mais escura que outra. Em outras palavras, o contraste é a diferença do brilho entre regiões claras e escuras de uma imagem.

Por meio do histograma da imagem, a visualização do contraste apresenta-se mais fácil de ser percebida. Se o histograma é mais fechado, afirma-se que a imagem possui baixo contraste, e se o histograma for mais aberto, a imagem possui alto contraste.

A técnica para o ajuste de contraste adotado consiste em aumentar ou reduzir a cobertura dos limites inferiores e superiores dos valores dos pixels de uma imagem. Em outras palavras, é como se esticasse ou encolhesse a distribuição dos pixels representados no histograma de uma imagem.

Para realizar esta operação, basta multiplicar os pixels de uma imagem por um valor decimal maior que zero. Quando o valor de ajuste for menor que um, a imagem será ajustada para que tenha contraste baixo, ou seja, a distribuição dos pixels é aproximada, e quando o valor for maior que um, o contraste será alto, aumentando a distância entre os valores dos pixels (GONZÁLEZ; WOODS, 2008).

Para realizar esta operação, utiliza-se um algoritmo que contém as seguintes etapas:

- Calcular a média dos pixels da imagem;
- Subtrair o valor da média do valor de cada pixel, provocando o deslocamento do histograma da imagem para a esquerda;
- Multiplicar o valor de cada pixel por um escalar. Se o escalar for maior que 1, a largura do histograma será aumentado, caso contrário, será encolhido;
- Somar a média dos valores dos pixels aos novos valores obtidos na etapa anterior;
- Ajustar os pixels que extrapolaram a escala do espaço de cor utilizado, tanto no limite inferior quanto no superior.

Utilizando a mesma imagem da Figura 13, a Figura 14 mostra os ajustes de contraste alto e baixo. Nota-se visualmente que o efeito do ajuste do contraste altera o histograma da imagem de forma que ele fique encolhido (contraste baixo) ou esticado (contraste alto).

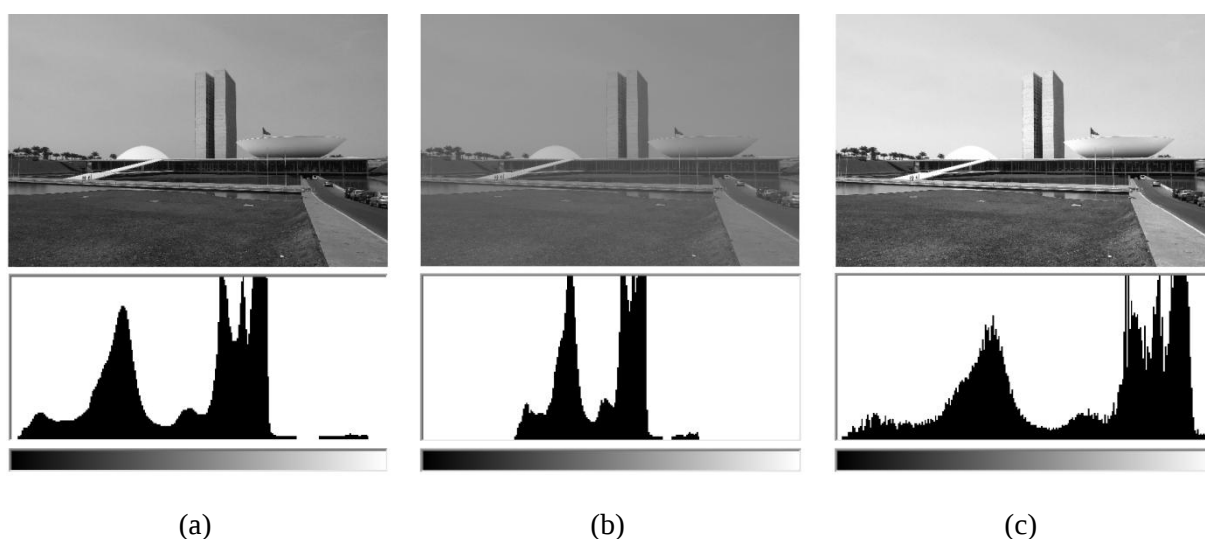


Figura 14 – Ajuste de contraste; (a) Imagem original; (b) Imagem com baixo contraste; (c) Imagem com alto contraste

A percepção visual do ajuste de contraste traz a ideia de opacidade em uma imagem. Quanto menor o contraste, mais opaca é a iluminação da imagem. Dentro do

objetivo proposto pela *engine* do ambiente virtual, os ajustes de brilho e contraste devem ser realizados com o intuito de destacar o objeto a ser rastreado.

Se o contraste for ajustado de forma baixa, os pixels da imagem ficarão muito próximos dentro da distribuição espacial da imagem. A configuração ideal é aquela em que se realizam os ajustes de forma que o histograma fique centralizado (pelo brilho) e esticado (pelo contraste) sem que ocorra a extrapolação dos pixels tanto no limite inferior quanto no limite superior.

5.3 Redução da Paleta de Cores

As imagens digitais no espaço de cor RGB são representadas geralmente por uma paleta de cor de 24 bits, no qual são destinados oito bits para cada componente de cor. Dependendo da forma de visualização da imagem, cores muito próximas não são fáceis de serem distinguidas.

Há técnicas para a redução da magnitude do espaço de cor que torna a distância entre as cores maior e reduz o tamanho de bits de um pixel. A técnica utilizada neste projeto consiste primeiramente em definir o número de cores por canal RGB e depois realizar o arredondamento de uma cor para que ela esteja dentro de um intervalo definido.

Cada dimensão do espaço de cor RGB é representada no intervalo de números inteiros de 0 a 255. O algoritmo de redução do espaço de cor consiste em manter o mesmo intervalo, todavia, o incremento do valor da cor ao invés de ser unitário, é estabelecido por uma aproximação ajustada por meio do número de cores por canal.

Para realizar esta aproximação, foi utilizada a equação (9). Essa equação representa o arredondamento do valor de cada componente da cor pelo tamanho do intervalo a ser definido (ARVO, 1991).

$$v' = \text{quociente}(v/n) * n \quad (9)$$

onde v' é o valor da cor ajustada, v é o valor da cor original e n é o número de cores existentes no intervalo.

Na Figura 15, pode-se visualizar uma imagem com 256 cores por canal e outra com 12 cores por canal, bem como a distribuição dos pixels das imagens dentro do espaço RGB.

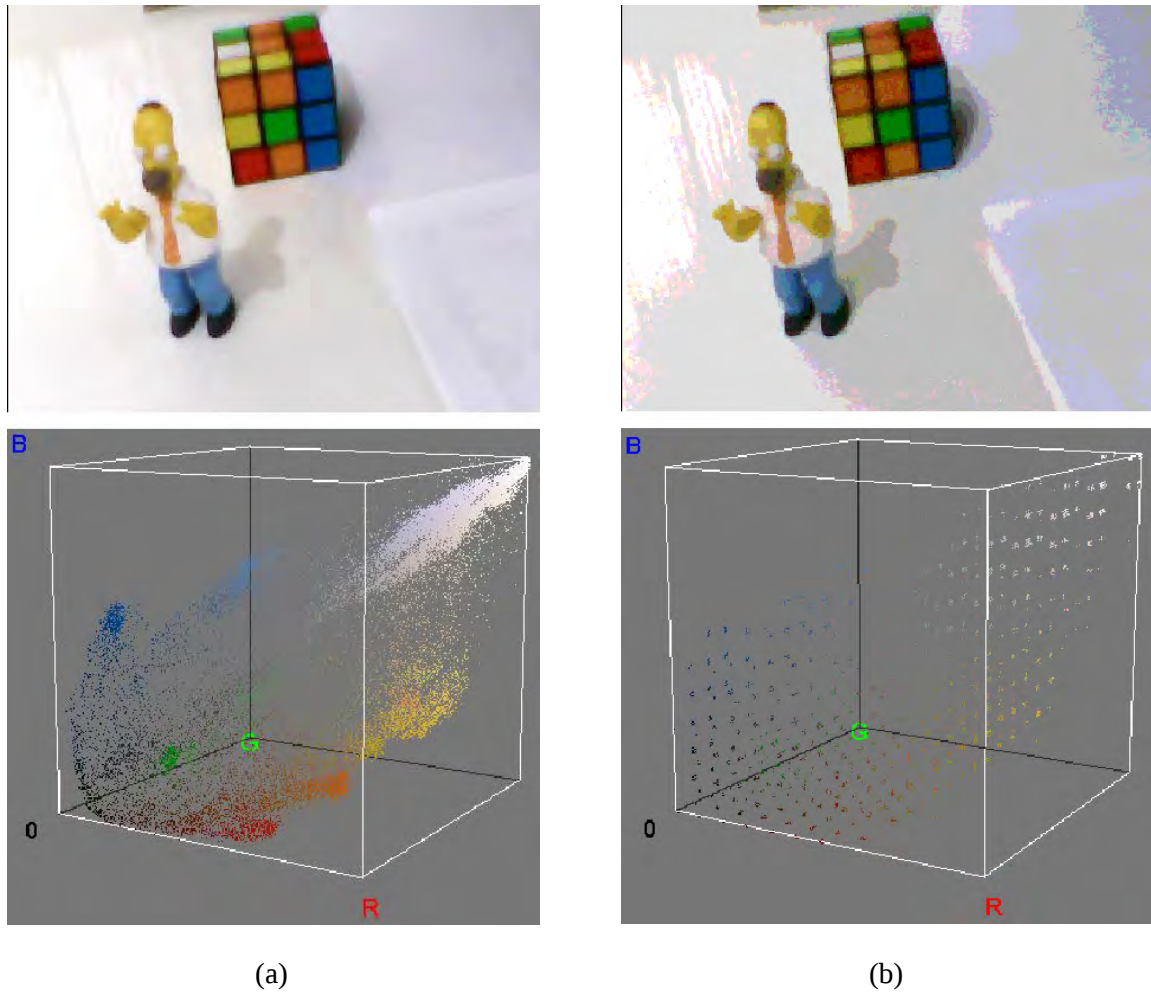


Figura 15 – Redução da paleta de cores. (a) 256 cores por canal; (b) 12 cores por canal

Ao reduzir a paleta de cores da imagem, a distribuição das cores fica mais esparsa, tanto na própria imagem quanto no cubo RGB. Este processo de redução é um fator que melhora o desempenho dos algoritmos de segmentação, pois os métodos de segmentação escolhidos são baseados na análise pixel a pixel de uma imagem.

5.4 Segmentação

O processo de segmentação tem como objetivo analisar uma imagem de entrada e separar os elementos que compõem o primeiro plano do segundo plano. Em outras palavras, a imagem deve ser tratada de modo que os objetos de interesse fiquem em primeiro plano, enquanto o restante da imagem seja desprezado.

Os métodos de segmentação de imagens estudados neste projeto transformam a imagem para o formato binário, onde 0 (ou falso) representa a cor preta e 1 (ou verdadeiro) a cor branca. A imagem binarizada pode conter vários *blobs* que representam as regiões de interesse (GARCIA *et al.*, 2007).

Os *blobs* correspondem a regiões da imagem nas quais aparecem pixels do mesmo valor adjacentes entre si. Na Figura 16, há a presença de três *blobs*: a região pontilhada de vermelho identifica a região de cada um dos *blobs*.

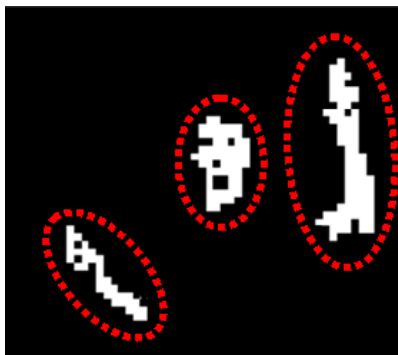


Figura 16 – Exemplo da ocorrência de *blobs* em uma imagem

Sendo assim, o que se espera da operação de segmentação é um mecanismo de entrada e saída, no qual a entrada é a imagem pré-processada e a saída é a imagem binarizada de forma que os *blobs* sejam os objetos de interesse.

Foram avaliados cinco métodos para a segmentação da imagem: subtração de *background* (CHEUNG; KAMATH, 2004), filtro por histograma de cores (FREITAS *et al.*, 2007), normalização de cores (SEDLÁČEK, 2004), indexação de tons e tonalidade máxima e mínima. Todos esses métodos são fáceis de serem codificados e possuem uma complexidade computacional baixa. Os métodos de segmentação são representados por uma função de binarização $b(x)$ que tem x como o pixel analisado e como resposta valores 0 ou 1.

Os métodos estudados, com exceção do de subtração de *background*, necessitam de efetuar um treinamento para que o algoritmo obtenha as características dos objetos de interesse presentes na imagem. O treinamento é realizado toda vez que o sistema é executado e consiste em destacar uma região da imagem adquirida pela câmera para a análise.

5.4.1 Subtração de Background

O método de subtração de *background* possui uma característica peculiar em relação aos outros métodos avaliados. É necessário que uma variável temporal seja adotada, pois a função de binarização da imagem faz referência aos instantes de tempo anterior e atual na aquisição de imagens sequenciais.

Cada quadro é representado como matrizes algébricas e em cada um o valor de cada elemento da matriz corresponde ao valor da cor do pixel. A restrição que o algoritmo de

subtração de background possui é que o frame anterior ($t - 1$) seja sempre armazenado, pois este faz parte do processo de binarização.

Antes de aplicar a função $b(x)$, é realizada a subtração entre as matrizes do quadro atual e do quadro anterior. O resultado desta operação é ilustrado na Figura 17:



Figura 17 – Operação de subtração entre o quadro atual e o quadro anterior

Nota-se que uma grande parte da imagem ficou com a cor preta (representada pelo valor zero) pelo fato dos pixels manterem-se com o mesmo valor em algumas regiões das imagens anterior e atual. A região que apresenta um vulto é o local onde ocorreu diferença entre o quadro do instante atual e o anterior.

À imagem resultante da subtração das imagens é aplicada a função de binarização $b(x)$ sugerida pela equação (10). O valor da variável *threshold* é definido como um parâmetro de entrada. Esta variável indica o nível de tolerância que separa os pixels branco e preto:

$$b(x) = \begin{cases} 0, & \text{se } x \leq \text{threshold;} \\ 1, & \text{caso contrário.} \end{cases} \quad (10)$$

A Figura 18 apresenta o resultado da função de binarização (10) aplicada nos pixels da imagem resultante da subtração entre a imagem atual e a anterior da Figura 17.



Figura 18 – Segmentação da imagem pelo método de subtração de background

O resultado deste método identifica onde ocorre a movimentação, porém ele não serve para identificar objetos que permanecem estáticos de um frame para o outro. Se o cenário não se altera durante uma sequência de frames, o resultado sempre será uma matriz com valores zero.

As próximas subseções apresentam métodos que realizam a segmentação de forma independente do tempo. Para resultados mais consistentes, indica-se que os objetos de interesse tenham as cores diferentes do restante do ambiente.

5.4.2 Filtro por Histograma de Cores

O método de segmentação pelo filtro de histograma de cores consiste em calcular o histograma médio da imagem de cada componente da cor RGB. Inicialmente, é necessário executar o processo de treinamento que calcula o histograma médio de uma parte da imagem, a área de treinamento, durante meio segundo. A área de treinamento e os histogramas médios são apresentados na Figura 19.

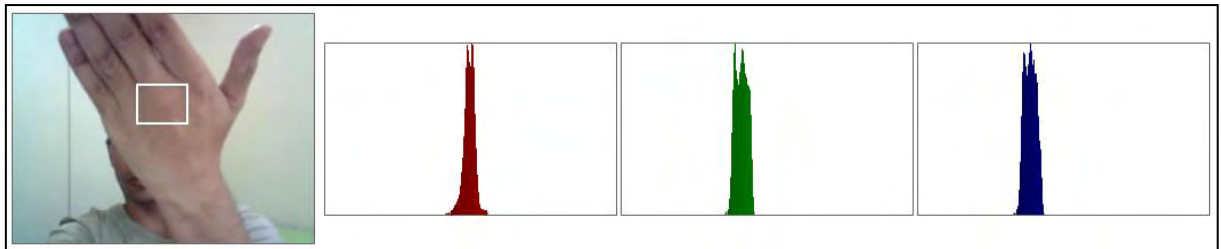


Figura 19 – Histograma médio obtido no treinamento de cada componente RGB

A partir do histograma médio, é definido um filtro que será utilizado na função $b(x)$, no qual o valor do pixel é desmembrado em cada componente de cor RGB, variando de 0 a 255. A função de binarização utilizada neste método define o pico do histograma como uma região central e é parametrizado um limiar de tolerância ao redor dessa região central. A aceitação do limiar de tolerância se dá quando todas as componentes de cor estão inseridas na faixa de tolerância. Este comportamento é descrito pela equação (11):

$$b(x) = \begin{cases} 1, & \text{se } (R - T) < r < (R + T) \text{ e } (G - T) < g < (G + T) \text{ e } (B - T) < b < (B + T); \\ 0, & \text{caso contrário.} \end{cases} \quad (11)$$

onde as variáveis R , G e B correspondem aos valores dos picos do histograma. As variáveis r , g e b são as componentes de cor do pixel x e a variável T é o limiar de tolerância do filtro.

A Figura 20 apresenta um exemplo para essa função de binarização utilizando o histograma médio apresentado na Figura 19. Neste exemplo foi utilizado o limiar de tolerância T igual a 30.

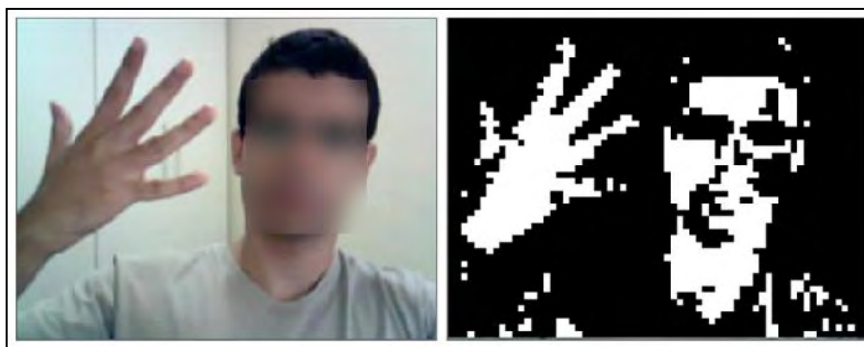


Figura 20 – Segmentação efetuada pelo método de filtro por histograma de cores

Esse método consegue executar o processo de segmentação próximo do ideal, pois há a presença de falso positivo em alguns locais da imagem segmentada. Uma possível solução é desfocar a imagem para que pequenos *blobs* desapareçam. O próximo método apresentado consiste em efetuar a segmentação pelas cores normalizadas.

5.4.3 Normalização de Cores

O método de normalização de cores consiste em obter o coeficiente de cada componente RGB de um pixel. Tal coeficiente é capaz de abstrair questões do ambiente como variação de iluminação, contraste e brilho mal configurados em relação ao ambiente externo (SEDLÁČEK, 2004).

Esse método possui um processo de treinamento semelhante ao método de filtragem por histograma de cores, porém, ao invés de obter o histograma médio, são calculados os coeficientes médios de uma determinada região da imagem. A Figura 21 mostra um exemplo de treinamento com os coeficientes médios obtidos na região de treinamento da imagem.

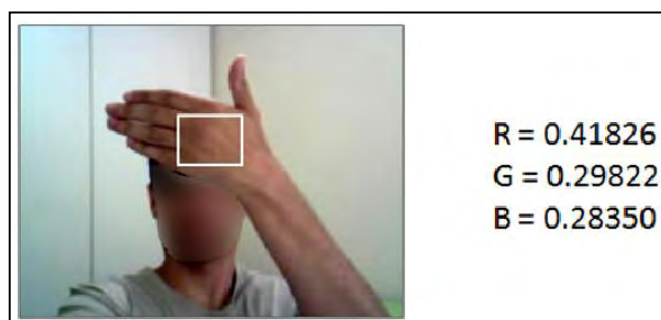


Figura 21 – Coeficientes médios calculados no treinamento

A função de binarização também é semelhante à função do método anterior. São calculados os coeficientes de cada pixel e é verificado se os coeficientes aproximam-se dos

coeficientes médios por meio de um limiar de tolerância T . A equação (12) mostra como funciona a tomada de decisão para os pixels analisados.

$$b(x) = \begin{cases} 1, & \text{se } (R' - T) < r' < (R' + T) \text{ e } (G' - T) < g' < (G' + T) \text{ e } (B' - T) < b' < (B' + T); \\ 0, & \text{caso contrário.} \end{cases} \quad (12)$$

onde as variáveis r' , g' e b' são os coeficientes do pixel analisado e R' , G' e B' são os coeficientes médios calculados no treinamento e T é o limiar de tolerância.

A Figura 22 apresenta o resultado do processo de segmentação utilizando o treinamento configurado no exemplo da Figura 21. Neste exemplo foi utilizado o limiar de tolerância T igual a 0,03.

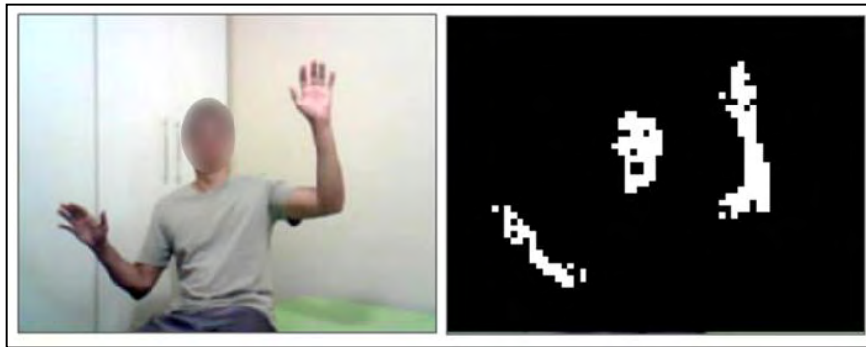


Figura 22 – Segmentação efetuada pelo método de normalização de cores

O nível de ruído apresentado nesse método é baixo. Analisando a Figura 22, observa-se que, por questões de iluminação do ambiente, o braço esquerdo da pessoa na imagem está mais claro que o braço direito. Já foi visto que o resultado do algoritmo de segmentação por RGB normalizado comprova o fato de que o método abstrai as características de iluminação da imagem processada (VEZHNEVETS;SAZONOV; ANDREEVA, 2003).

Os próximos métodos apresentados utilizam o espaço de cor HSV para realizar as operações que determinam a segmentação da imagem. A estratégia principal desses métodos é de trabalhar com a tonalidade das cores e descartar as componentes de saturação e iluminação.

5.4.4 Indexação de Tons

O método de indexação de tons consiste em armazenar as tonalidades dos pixels contidos na região de treinamento. O treinamento é realizado da mesma forma que os métodos de Normalização de Cores e Filtro por Histograma de Cores, porém o treino realiza a indexação da componente H (tonalidade) dos pixels analisados.

Conforme o treino é realizado, as componentes H dos pixels são armazenadas em um vetor booleano de tamanho 360. A cada iteração do treino é realizada a conversão da cor

do pixel analisado de RGB para HSV e, então, a posição do vetor correspondente ao valor da variável H, que tem o limite de 0 até 360, é marcada como verdadeiro.

Para evitar a adição de pixels que resultem em falso positivo na segmentação, pixels com a saturação (S) abaixo de 20 e o valor (V) abaixo de 20 são descartados. O motivo dessa escolha se dá pelo fato que cores com esses valores para S e V vão se misturando na região central do cone HSV apresentando a coloração cinza.

Por exemplo, uma cor da tonalidade azul com baixa saturação é visualmente igual a uma cor da tonalidade verde com baixa saturação. O algoritmo de conversão de RGB para HSV não capta essa nuance do espaço de cor, resultando no azul e verde de baixa saturação (cinza) como uma mesma cor HSV.

Na Figura 23 é mostrada uma região utilizada no treinamento, distribuída no espaço de cor HSV. Nota-se que as tonalidades da região de treinamento estão concentradas em poucos valores H, ocupando uma pequena área do círculo representado por H e S.

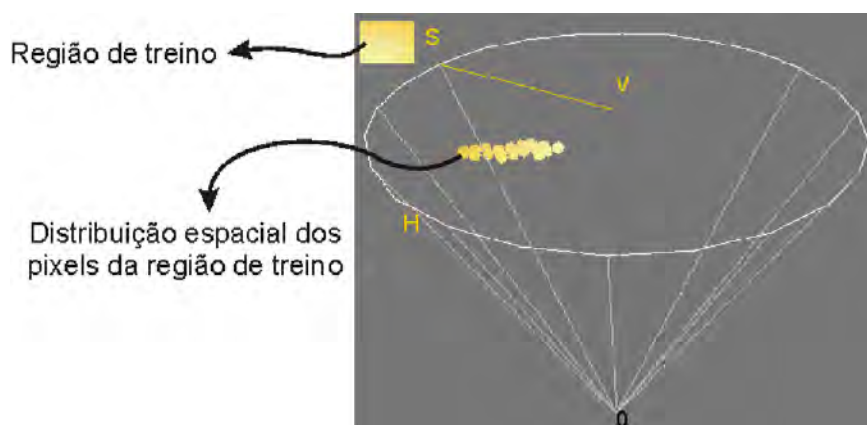


Figura 23 – Distribuição espacial dos pixels da região de treino

O resultado deste método de segmentação é apresentado na Figura 24. Nesta captura, foi utilizada a mesma região de treinamento da Figura 23. As partes mais escuras da esfera não entraram no critério de aceitação, pois apresentam tonalidade diferente das que foram armazenadas no treinamento.

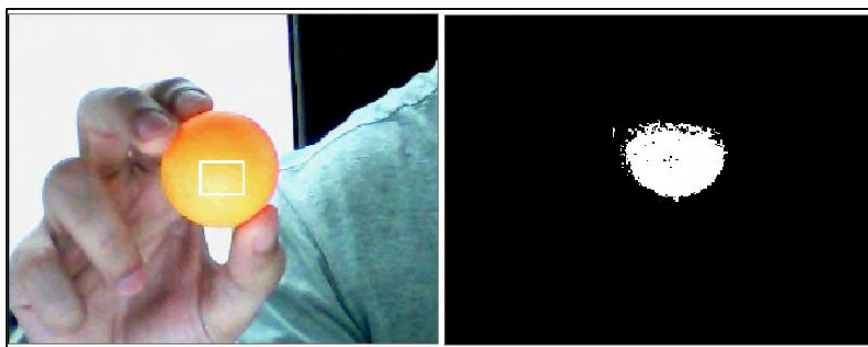


Figura 24 – Segmentação utilizando indexação de tons. A região de treinamento é o retângulo branco na imagem à esquerda.

A segmentação realizada por este método possui um critério de aceitação muito rigoroso. Se há pixels com tons intermediários aos do que foram indexados, os mesmos são declarados como falsos.

5.4.5 Tonalidade Mínima e Máxima

Com o objetivo de maximizar o critério de aceitação do método de tons indexados, foi definida uma estratégia para que a aceitação dos pixels englobe uma região máxima e mínima da componente H.

Quando o treinamento é realizado, ao invés de armazenar o vetor com as tonalidades máximas e mínimas, são guardados apenas os valores mínimo e máximo de H. Este intervalo será utilizado para determinar o critério de aceitação (quando o pixel analisado estiver dentro desse intervalo) e rejeição.

A representação gráfica da região de aceitação assemelha-se a uma fatia do cone HSV que abrange da tonalidade mínima até a máxima. Utilizando o mesmo treinamento ilustrado na Figura 23, a Figura 25 mostra a representação espacial da região de treinamento do método de segmentação por tonalidade mínima e máxima.

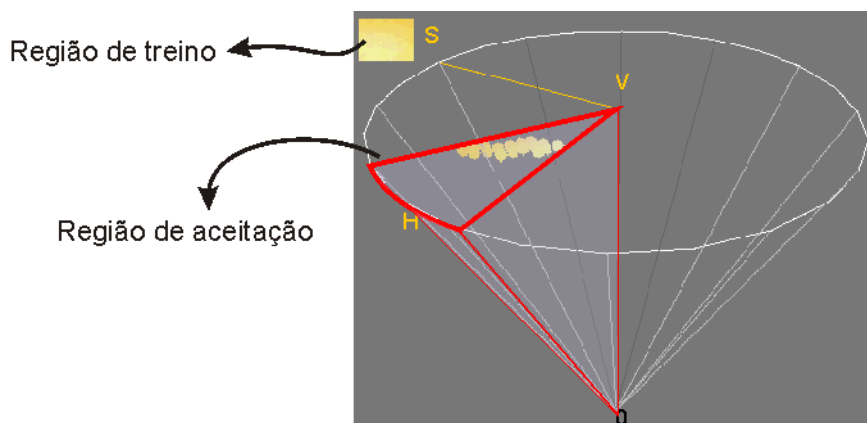


Figura 25 – Representação espacial da região de treinamento do método de segmentação por tonalidade mínima e máxima

O resultado deste método de segmentação é apresentado na Figura 26. Alguns pixels falsos positivos passaram pelo critério de aceitação. Para evitar este tipo de problema, seria necessário o uso de um objeto com a tonalidade bem distinta em relação ao restante da imagem.

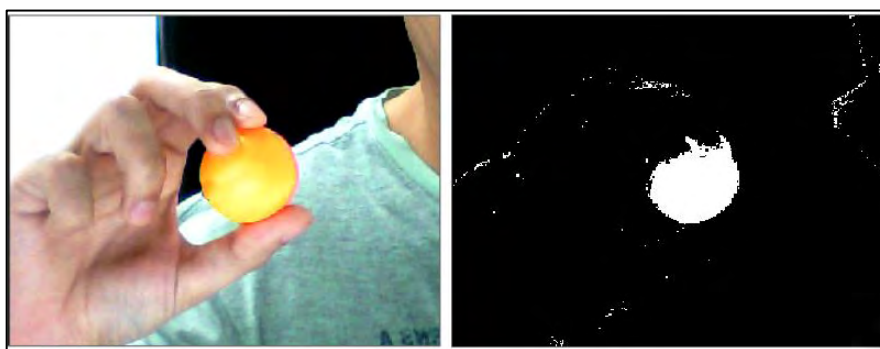


Figura 26 – Segmentação utilizando tonalidade mínima e máxima

A comparação entre os métodos e a escolha de qual método é mais apropriado em diversas situações serão apresentadas no Capítulo 7.

5.5 Dilatação

O processo de dilatação consiste em realçar um *blob* da imagem, fazendo com que seus pixels vizinhos façam parte do mesmo *blob*. Quando se trata de dilatação, é escolhida a cor a ser realçada, que geralmente é a cor branca.

A dilatação corresponde à soma lógica (OR) de um grupo de pixels da imagem representado pela matriz A e pelo elemento estruturante B (fig. 27). A matriz A corresponde ao pixel que está sendo analisado juntamente com sua vizinhança, resultando em uma matriz 3×3 . O elemento estruturante também possui a dimensão 3×3 e corresponde a uma máscara que define a estratégia da dilatação (DOUGHERTY; LOTUFO, 2003).

O algoritmo de dilatação percorre a imagem até encontrar um pixel com a cor selecionada para o realce. Quando a cor é encontrada, é obtida a matriz A e é efetuada a operação lógica OR com o elemento estruturante B .

A Figura 27 ilustra uma iteração do algoritmo de dilatação. A imagem possui a dimensão de 9 x 9 pixels e a dilatação é aplicada nos pixels de cor cinza. O pixel selecionado está na oitava coluna da terceira linha e a área delimitada de vermelho representa a matriz A .

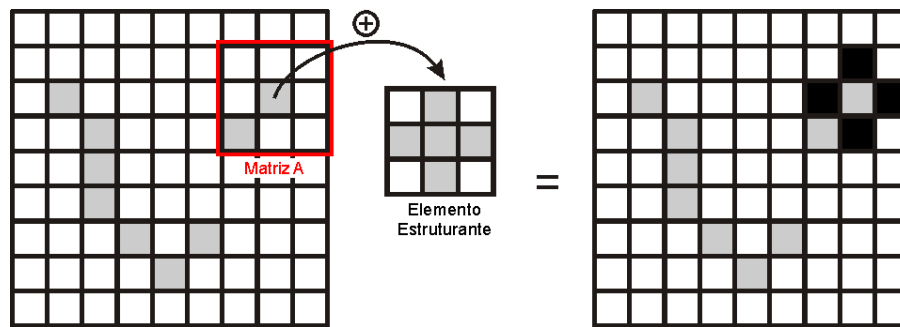


Figura 27 – Iteração do algoritmo de dilatação

Os pixels representados pela cor preta são resultantes da operação lógica com a matriz do elemento estruturante. Na continuação do algoritmo seria escolhido outro pixel cinza para compor a matriz A e a operação lógica com o elemento estruturante seria aplicada novamente, até que fossem utilizados todos os pixels da cor cinza. A Figura 28 ilustra como ficaria a imagem no final da execução do algoritmo de dilatação.

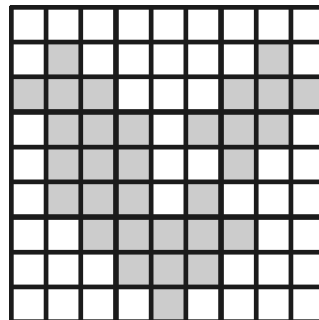


Figura 28 – Imagem após a execução do algoritmo de dilatação

Foi aplicado o algoritmo de dilatação utilizando a imagem segmentada da Figura 24 (fig. 29). Nota-se que pequenos ruídos foram descartados e a imagem dilatada apresenta um único *blob*.



Figura 29 – Resultado do algoritmo de dilatação aplicado à imagem segmentada da Figura 24;
(a) Imagem original; (b) Imagem dilatada.

Como resultado do processamento completo, o algoritmo de dilatação permite utilizar objetos pequenos e também que o usuário esteja posicionado a uma distância maior da câmera. Quanto mais longe a pessoa está em relação à câmera, maior é a amplitude do movimento capturado, permitindo que ocorra a execução de tarefas mais complexas pela pessoa.

6 MATERIAIS E MÉTODOS

O presente estudo foi realizado no Laboratório de Bioengenharia e Biomecânica da Universidade Federal de Goiás, Campus II, Goiânia, no período de 2010 a 2012.

O trabalho foi dividido em etapas teóricas e práticas. A revisão de literatura foi realizada ao longo do projeto, sendo a maior parte efetuada na definição do escopo deste trabalho.

Após a definição do escopo do projeto, foi dado o início ao desenvolvimento do primeiro protótipo da *engine* utilizada no ambiente virtual. Foram necessários alguns estudos sobre visão computacional e IHC.

A linguagem de programação utilizada no desenvolvimento da *engine* foi a *Action Script 3* para produzir o produto final na plataforma *AIR 3.1*, ambas da *Adobe Systems Incorporated*. A escolha desta plataforma de desenvolvimento se deu pela facilidade de trabalhar com vídeos e imagens e também por ser multiplataforma (a aplicação pode ser executada em vários sistemas operacionais desktop). A aplicação também pode ser convertida para *Adobe Flex*, permitindo que seja executada em navegadores que tenham o plugin *Flash Player* instalado, facilitando sua implantação quando a aplicação for testada em ambientes externos.

A configuração do hardware utilizado para desenvolvimento e testes no laboratório foi um *laptop* com processador *Intel Core 2 Duo* com 4 GB de memória RAM. A *webcam* é da fabricante *OmniVision Technologies* com resolução de 2 Megapixels, frequência de 30 frames por segundo integrada ao *laptop* (OMNIVISION TECHNOLOGIES).

O módulo do ambiente virtual foi desenvolvido desacoplado da *engine*, pois é de responsabilidade do ambiente virtual apenas os eventos e ações relacionadas ao domínio do jogo. O domínio do jogo consiste nas regras de negócio, efeitos audiovisuais e informações de saída.

A modelagem do ambiente virtual foi elaborada a partir de visitas técnicas realizadas no Centro de Reabilitação e Readaptação Dr. Henrique Santillo (CRER), localizado no Setor Vila Nova, em Goiânia, na área de terapia ocupacional e fisioterapia. Estas visitas foram importantes para o conhecimento do problema em recuperação funcional para o qual esse projeto foi desenvolvido, bem como do ambiente de terapia.

O maior contato no CRER foi com pacientes que sofreram Acidente Vascular Encefálico (AVE) e possuem a função motora prejudicada, sobretudo nos membros superiores. O movimento dos membros superiores é, por definição, dividido em alcance, manipulação e preensão (SHUMWAY-COOK; WOOLLACOTT, 2007). O ambiente virtual

modelado consiste em incentivar as tarefas de alcance e preensão tanto contralateral (do lado oposto ao membro) quanto ipsilateral (do mesmo lado do membro lesado).

Após concluído o protótipo do ambiente virtual, foram realizados testes no laboratório em indivíduos normais com o intuito de levantar informações sobre as restrições necessárias para o funcionamento da ferramenta. A descrição do desenvolvimento da ferramenta é dada primeiramente com foco na *engine* e depois é mostrado como foi desenvolvido o ambiente virtual.

7 ENGINE DO AMBIENTE VIRTUAL

O desenvolvimento do ambiente virtual consiste primeiramente em projetar a *engine*, e em um segundo momento, desenvolver as aplicações que utilizam os recursos disponibilizados pela *engine*. A aplicação desenvolvida é classificada como um sistema de realidade virtual *desktop*, ou seja, todo o aparato para a interação do usuário até a visualização da resposta do jogo é realizado em computadores pessoais.

A *engine* do ambiente virtual proposto é responsável por fornecer as operações de reconhecimento e rastreamento de objetos de interesse por meio de *webcams*. O processamento das imagens é realizado conforme os quadros são capturados e as informações sobre a posição dos objetos de interesse são obtidas na mesma iteração.

No final da análise do quadro capturado, as informações sobre os objetos identificados são repassadas para o módulo de jogo que faz uso desta informação para atualizar as variáveis internas e prover interação com o usuário.

Portanto, a *engine* do ambiente virtual é responsável por realizar as operações de monitoração e visualização, restando as ações de controle sob responsabilidade dos módulos de jogo. A seguir, será descrito como foi desenvolvida a *engine* do ambiente virtual e também as discussões dos resultados obtidos ao longo de seu desenvolvimento.

7.1 Especificação

Os requisitos recomendados para o funcionamento da *engine* são, na sua maioria, transparentes para o usuário, pois alguns dos recursos não são visíveis na interface gráfica. Contudo, estes recursos são aspectos computacionais que garantem que o ambiente virtual terá a usabilidade e o desempenho satisfatório para o usuário.

Os requisitos não funcionais são:

- O processamento das imagens deve ser realizado de forma *online*, ou seja, a percepção do usuário deve ser de que a execução da ferramenta seja em tempo real, sem atrasos;
- A trajetorização é realizada pela segmentação dos objetos por meio da análise das cores extraídas no processo de treinamento;
- O treinamento dos métodos de trajetorização deve ser realizado a cada nova seção de jogo. Os métodos de trajetorização devem ser robustos ao ponto de não precisarem efetuar a recalibração na mesma seção de jogo;

- As configurações dos parâmetros de calibração da *engine* devem ser flexíveis para que o sistema funcione em vários tipos de ambientes;
- A usabilidade para os usuários envolvidos (pacientes e terapeutas) deve ser intuitiva (vide boas práticas de condução) (SILVA; SILVEIRA, 2008).

Por outro lado, os requisitos funcionais consistem em configurações realizadas de forma explícita nos ajustes dos parâmetros de pré-processamento, realização de treinamento (quando é exigido pelo método de segmentação) e a visualização da imagem original e processada em tempo real.

Sob o ponto de vista temporal, a *engine* é responsável pela aquisição da imagem da *webcam*, análise desta imagem e envio das informações da análise para os módulos externos por meio de eventos. A Figura 30 ilustra como é organizado o fluxo das tarefas realizadas pela *engine* e os ajustes e opções de configurações que podem ser utilizadas em cada uma das tarefas.



Figura 30 – Fluxograma do processamento realizado pela *engine*.

A principal característica da *engine* é analisar as imagens de entrada e ter como saída as informações cartesianas dos objetos de interesse. As tarefas listadas na Figura 30 são realizadas de forma sequencial, sendo o processo de segmentação considerado o gargalo no desempenho do processamento como um todo.

Há alguns fatores externos que influenciam diretamente na configuração dos parâmetros da *engine* como: a cor e tamanho do marcador, iluminação do ambiente, qualidade da *webcam* e distância do usuário em relação a *webcam*. As próximas subseções descrevem como foi construída cada tarefa da ferramenta e quais foram as melhores configurações obtidas dentro do ambiente de pesquisa.

7.1.1 Projeto

Do ponto de vista técnico, o projeto estrutural e a codificação da *engine* foi realizada de acordo com o paradigma de Orientação a Objetos (OO). Foram utilizados os diagramas da *Unified Modeling Language* (UML) de classe e de sequência para representar os mecanismos internos da *engine*.

Este tipo de representação facilita a compreensão de como os objetos interagem entre si e também como é realizada a comunicação entre as classes do sistema. A Figura 31 mostra o diagrama de classes da *engine*.

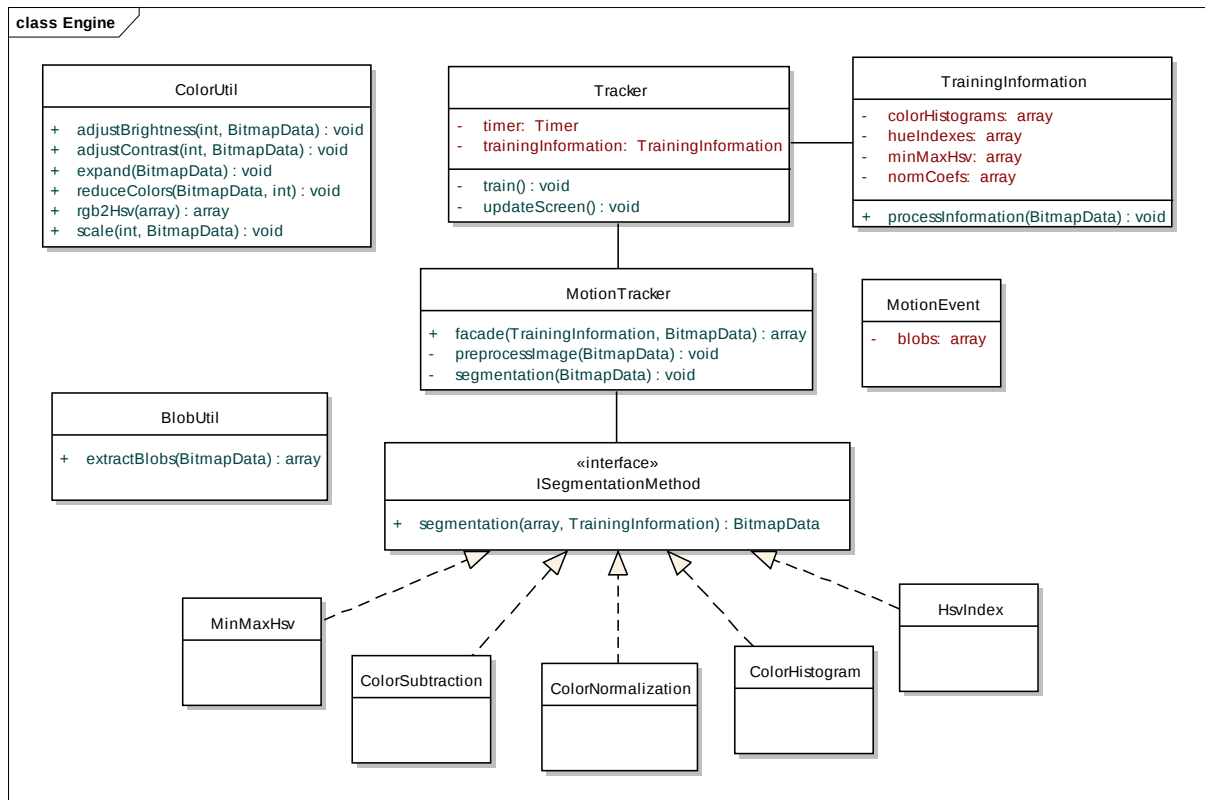


Figura 31 – Diagrama de classes da *engine*

A classe *Tracker* é a representação da camada de visão. É nela que estão os controles de tela, como menus, botões, mecanismos para entrada de parâmetros e visualização da informação. A frequência de aquisição da câmera é ajustada pelo atributo *timer*, que tem a característica de temporizador, ou seja, dispara um evento de x em x milissegundos.

Toda vez que o evento do temporizador é executado, a classe *Tracker* realiza a atualização da tela por meio do método *updateScreen*. As atualizações de tela consistem em alterar a imagem do quadro da *webcam* e também a posição dos pontos rastreados.

O processo de treinamento é efetuado pelo método *train*, que tem como produto o atributo *trainingInformation* da classe *TrainingInformation*. A classe *TrainingInformation* é responsável por armazenar as informações coletadas no treinamento de acordo com o método de segmentação escolhido.

Já a classe *MotionTracker* efetua todos os processos de pré-processamento, segmentação e análise da imagem. A classe *Tracker* interage com *MotionTracker* por meio do método *facade*, que tem como parâmetros de entrada a imagem adquirida e as informações do treino.

Os ajustes realizados no pré-processamento são implementados na classe utilitária *ColorUtil*, e a análise dos *blobs* na classe *BlobUtil*. Os métodos de segmentação devem

realizar a interface *ISegmentationMethod*. Esta interface contém a especificação do método de segmentação, criando a possibilidade do acoplamento de novos métodos de segmentação sem que seja necessário realizar alterações drásticas nas outras classes.

Depois que o processo de segmentação foi realizado e os *blobs* foram mapeados, a classe *Tracker* dispara um evento do tipo *MotionEvent* com as informações geométricas e espaciais dos *blobs*. Caso tenha um ambiente virtual carregado, este deve adicionar um *listener* para este tipo de evento.

A descrição do fluxo de informação do ponto de vista temporal é visualizada no diagrama de sequência da Figura 32. A atividade representada no diagrama de sequência é o rastreamento de objetos de interesse. As tarefas de aquisição e treinamento não são ilustradas neste diagrama, pois são ações que não envolvem comunicação entre objetos de classes diferentes.

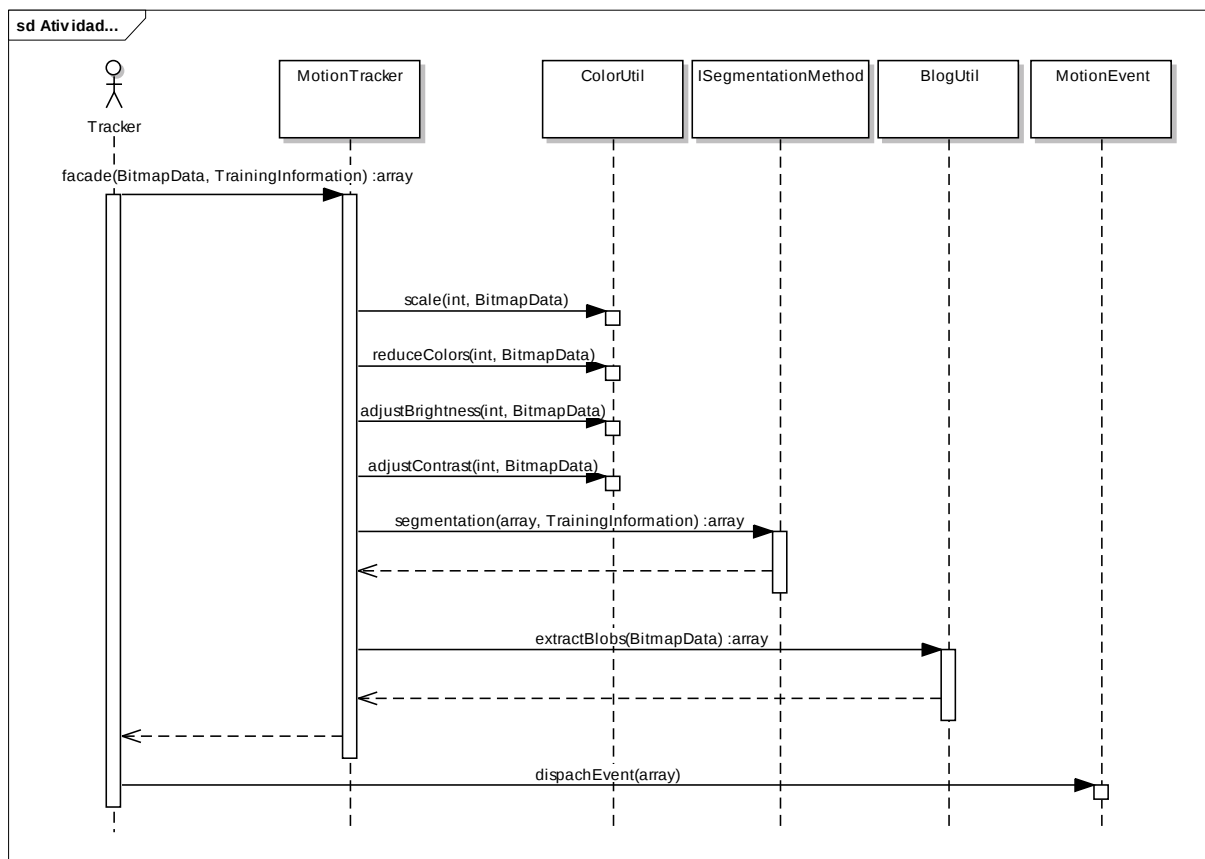


Figura 32 – Diagrama de sequência do processo de trajetORIZAÇÃO de objetos de interesse

No diagrama de sequência, o ator da operação é a própria classe *Tracker*, pois ela representa quem faz o controle e uso das atividades das outras classes. Pode-se observar como a execução do processo de rastreamento é realizada de forma síncrona. Este tipo de

visualização permitiu a identificação e o rastreamento de gargalos de desempenho durante a fase de desenvolvimento da *engine*.

As próximas subseções descrevem como foram desenvolvidas cada tarefa da *engine* e os principais resultados obtidos dentro do ambiente de testes.

7.1.2 Aquisição das Imagens

O processo de aquisição das imagens consiste em obter as imagens do mundo real por meio de um dispositivo de entrada, que nesse caso é a *webcam*. Primeiramente, deve ser definida a frequência de aquisição da imagem.

A configuração ideal seria de trabalhar sempre com a frequência máxima suportada pelo dispositivo. Porém, a quantidade de quadros por segundo tem impacto no desempenho da aplicação, pois o intervalo de tempo entre a aquisição de cada imagem pode se tornar menor que o tempo necessário para processar o ciclo de tarefas.

A definição da quantidade de quadros por segundo é relativa ao tempo que o computador gasta para executar o ciclo de atividades. No computador utilizado para o desenvolvimento, definiu-se, por meio de testes, a frequência de 15 quadros por segundo. Esta frequência estabelece o tempo limite para que o processamento das atividades aconteça de forma que não ocorram atrasos sensíveis ao usuário.

Os quadros capturados possuem as dimensões de 264 pixels de largura por 202 pixels de altura e estão no espaço de cor RGB. A cada momento em que um quadro é capturado, as tarefas relacionadas à análise da imagem são executadas. Logo após a conclusão das tarefas, o sistema fica ocioso até que a próxima imagem seja capturada.

7.1.3 Pré-Processamento

O pré-processamento da imagem ocorre antes do processo de segmentação. Esta etapa é importante pelo fato de realçar as características do objeto de interesse para a segmentação e também por efetuar ajustes que melhoram o desempenho do restante das atividades.

Nesse projeto foram utilizadas algumas técnicas, descritas no Capítulo 5, como elementos de pré-processamento. Primeiramente, foi realizado o ajuste da escala da imagem, diminuindo sua dimensão original. Em segundo lugar, foi reduzida sua paleta de cores e, por último, foram feitos ajustes de brilho e contraste.

A dimensão original das imagens capturadas é de 264 por 202 pixels. Foram testados alguns valores para a escala da imagem com o intuito de estabelecer um parâmetro

que não deixasse a imagem muito distorcida. Na Figura 33, observa-se uma imagem na dimensão original e redimensionada nas escalas de $1/2$, $1/4$ e $1/8$. Os objetos foram posicionados a uma distância de aproximadamente 30 centímetros da câmera.

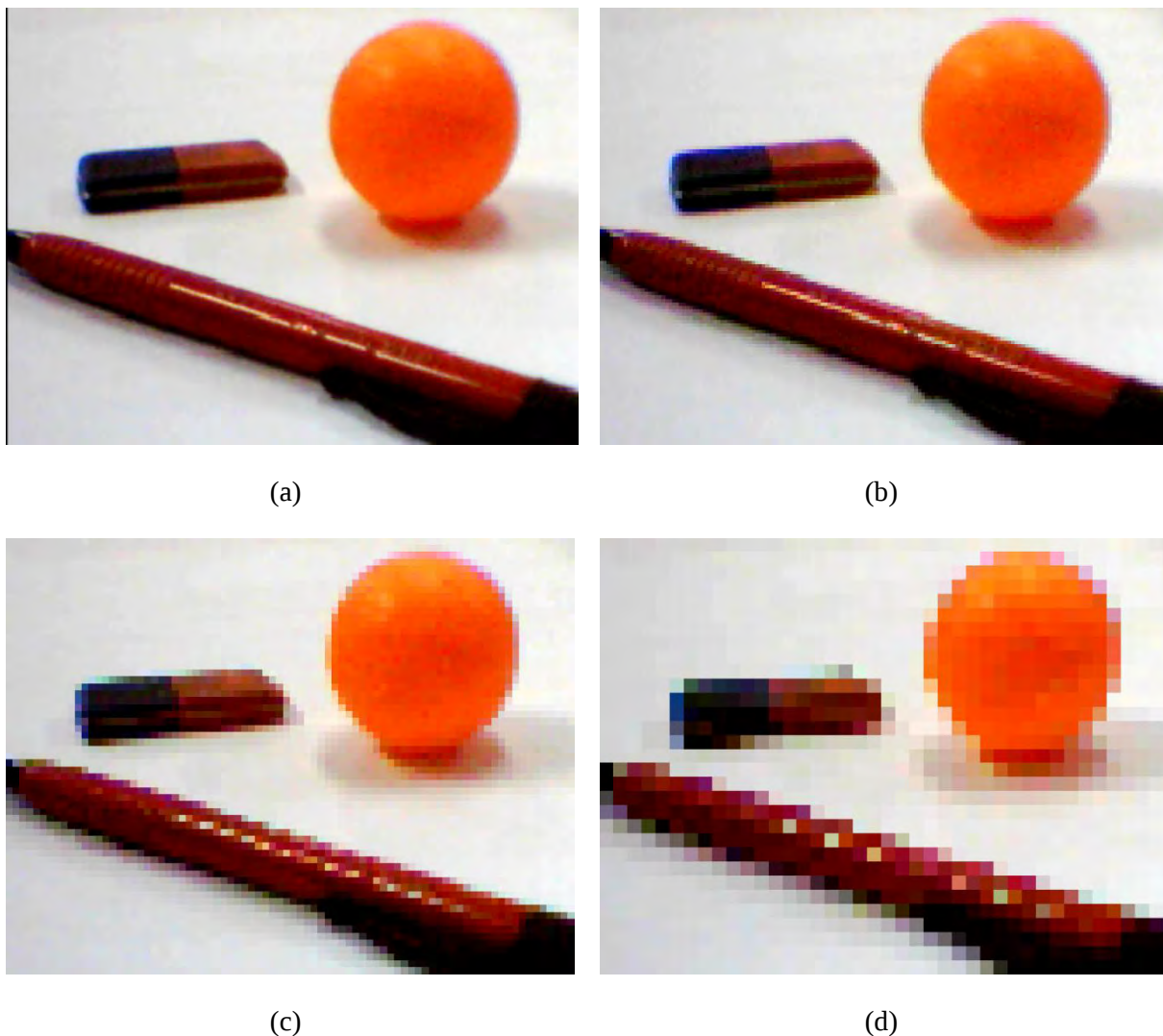


Figura 33 – Redução da escala da imagem. (a) Imagem original; (b) Escala de $1/2$; (c) Escala de $1/4$; Escala de $1/8$

A alteração da imagem possui impacto no tempo gasto para executar o restante do processo. Ao executar as outras etapas sem alterar a escala, há atraso no processamento das tarefas. Utilizando a escala de $1/4$, juntamente com a frequência de 15 quadros por segundo, a execução da *engine* ficou em tempo real do ponto de vista do usuário.

Na medida em que a escala é reduzida, a imagem vai se distorcendo e pequenos objetos são reduzidos a pequenos conjuntos de pixels até que desapareçam da imagem. Nos testes realizados na escala de $1/8$, os objetos mais distantes foram apagados da imagem, havendo uma maior restrição no raio de ação dentro do qual o usuário pode realizar as tarefas físicas.

Tanto a escala da imagem quanto a frequência de aquisição das imagens são dois parâmetros que tem maior impacto no desempenho da *engine*. Porém, estes parâmetros devem ser ajustados de forma que a usabilidade do sistema não seja prejudicada ao ponto de reduzir a percepção de movimento (frequência de aquisição baixa) e encurtar o raio de ação do usuário (escala da imagem baixa).

Outra técnica utilizada no pré-processamento da imagem foi a de redução da escala de cores, que consiste em aumentar a distância euclidiana entre as cores. Este método executa a redução utilizando a paleta de cores RGB, que é configurada por padrão para ter 256 níveis de cor por canal. Logo, o número total de cores no espaço RGB é de 256^3 cores distintas. Caso uma imagem seja ajustada para 8 cores por canal, o espaço RGB terá a quantidade de 8^3 cores.

Seguindo esta linha de raciocínio, o ajuste da quantidade de cores por canal deve ser realizado sem que as cores, inicialmente distintas, não se tornassem a mesma cor, pois o algoritmo de redução trabalha com arredondamentos (vide equação (9)).

A Figura 34 apresenta uma mesma imagem com diferentes configurações de número de cores por canal. As imagens apresentam 256, 16, 4 e 2 cores por canal, respectivamente.

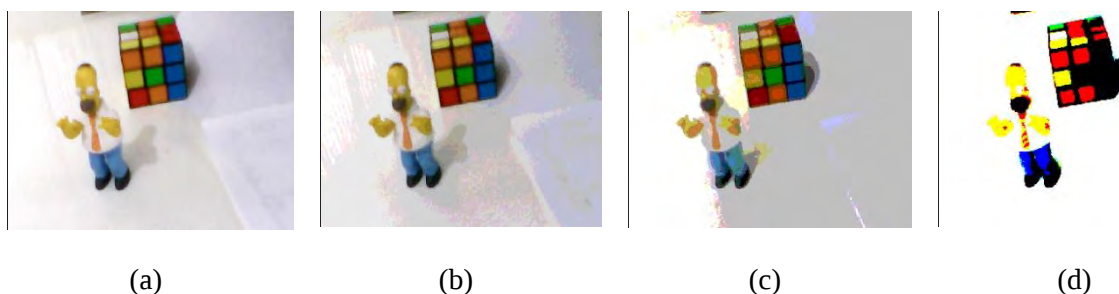


Figura 34 – Exemplo de redução da paleta de cores no espaço de cor RGB. (a) Imagem original; (b) 16 cores por canal RGB; (c) 4 cores por canal RGB; (d) 2 cores por canal RGB

Nota-se que utilizando a configuração de 4 cores por canal, a distinção entre as cores na imagem fica comprometida: pixels que eram diferentes (por exemplo, a sombra do boneco e a superfície onde ele está posicionado) tornaram-se iguais. Por outro lado, a unificação de pixels diferentes na configuração de 16 cores por canal mostrou-se tolerável para o processo de segmentação e esta foi a configuração utilizada como padrão no sistema.

Os últimos ajustes realizados na etapa de pré-processamento são os de brilho e contraste. Conforme visto anteriormente, estes dois ajustes podem atuar de forma que a distribuição das cores de uma imagem fique mais uniforme e também podem ser utilizados

para realçar determinados elementos de uma imagem. Neste caso, os ajustes de brilho e contraste são utilizados para realçar os objetos de interesse.

Os ajustes de contraste e brilho, demonstrados na Figura 35, foram realizados para destacar a mão dos outros elementos da imagem. Há ambientes em que o plano de fundo possui contraste em relação ao objeto de interesse por natureza. Nestas situações, os ajustes de brilho e contraste devem ser mínimos para que as cores não saturam, ou seja, para que não ocorra o deslocamento dos pixels para os limites inferiores e superiores (cores preta e branca) do espaço de cor. Por padrão, a *engine* tem a configuração inicial de brilho e contraste neutra.

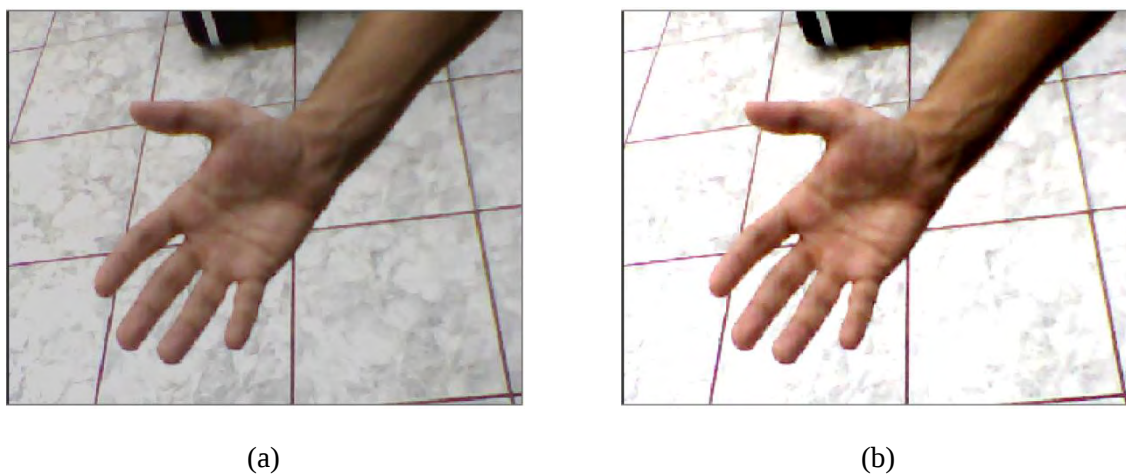


Figura 35 – Demonstração de ajuste de brilho e contraste. (a) Imagem original; (b) Imagem ajustada

As tarefas realizadas no pré-processamento preparam a imagem para a segmentação, que é responsável por separar elementos de interesse contidos em uma imagem digital. As próximas subseções discutem os resultados dos métodos de segmentação e como as imagens segmentadas foram analisadas.

7.1.4 Segmentação

Foram estudados cinco métodos de segmentação, sendo que estes métodos podem ser escolhidos no software via menu. O usuário tem a liberdade de escolher o método mais adequado ao tipo de marcador que está sendo utilizado por ele.

A escolha do método de segmentação acaba sendo subjetivo, pois não há um método que será bom sempre em todas as ocasiões. Devem ser levadas em conta as seguintes características externas: iluminação, distância do usuário, tamanho, cor e opacidade do marcador.

Foram realizados alguns testes nos métodos de segmentação com marcadores visuais de diferentes características. A cor do marcador é importante para a diminuição da

ocorrência de falso positivo e é aconselhável utilizar marcadores que possuem a cor distinta do restante dos elementos da imagem.

A Figura 36 apresenta uma comparação do resultado obtido nos métodos de segmentação. As segmentações foram realizadas utilizando a mesma região de treinamento, que na imagem está destacada pelo retângulo branco. As imagens obtidas pelo processo de segmentação estão sem qualquer tipo de tratamento. Posteriormente, é aplicado o método de dilatação para que os *blobs* fiquem destacados.

Tanto o método de normalização de cores quanto o de filtro por histograma de cores realizaram a segmentação da forma esperada. Já o método de tonalidade mínima e máxima apresentou alta ocorrência de falso positivo, enquanto que os pixels desejados não passaram pelos critérios de aceitação no método de indexação de tons.

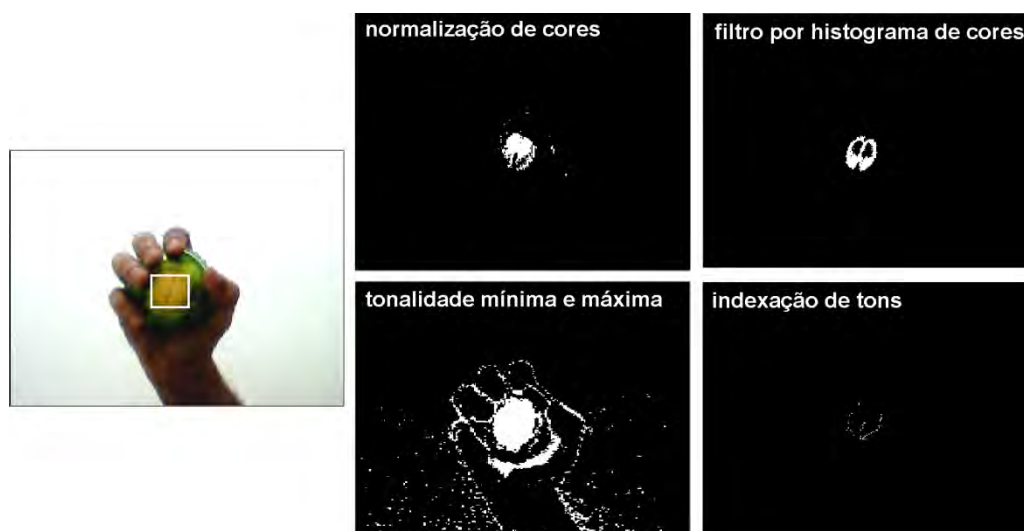


Figura 36 – Comparação entre métodos de segmentação utilizando um objeto de cor verde

O método de subtração de *background* não foi listado na comparação pelo fato de ser dependente do tempo e independente do uso de marcadores e questões de iluminação, fornecendo sempre um mesmo resultado, desde que haja diferença entre os dois quadros de imagem a serem utilizados na análise.

Diferente do resultado mostrado na Figura 36, o método de indexação de tonalidade apresentou melhores resultados no teste apresentado na Figura 37. No método de tonalidade mínima e máxima a área de cobertura dos tons abrangeram a esfera de cor laranja e também parte da mão. Os métodos de normalização de cores e filtro por histograma de cores também apresentaram resultados satisfatórios.



Figura 37 – Comparação entre métodos de segmentação utilizando um objeto de cor laranja durante o treinamento.

Pode-se afirmar que não há um espaço de cor e nem um método de segmentação melhor que outro. Alguns podem ser mais eficientes em alguns casos e outros não (ZARIT;SUPER; QUEK, 1999). Cabe ao pesquisador saber escolher o método de segmentação mais apropriado a ser aplicado no objeto de estudo (SHIN;CHANG; TSAP, 2002).

Para o usuário da *engine*, a escolha do método de segmentação será, na maioria dos casos, de forma empírica, pois não é de interesse do público alvo adquirir o conhecimento técnico sobre o funcionamento interno do dispositivo.

7.1.5 Análise da Imagem

Com a imagem segmentada é necessário realizar a separação dos *blobs*. O processo de análise da imagem consiste em armazenar as informações geométrica e espacial de cada *blob* em um vetor de informações. A partir destas informações, o módulo do ambiente virtual poderá atualizar o cenário e os elementos que interagem entre si.

As informações dos *blobs* estão organizadas no seguinte formato: coordenadas do centro, largura e altura. Estes valores são informados em uma escala percentual de 0 a 100, pois as dimensões do quadro da imagem capturada são sempre as mesmas. Todavia a área de trabalho no qual o ambiente virtual é projetado varia de acordo com o tamanho do monitor. Quando o ambiente virtual recebe as informações dos *blobs*, ocorre a conversão da escala relativa para os valores absolutos de dimensão e posicionamento.

Na tecnologia utilizada para a implementação da *engine* e do ambiente virtual, o método *getColorBoundsRect* da classe *BitmapData* realiza uma busca na imagem e determina

uma área retangular que cobre todos os pixels de uma cor selecionada, que neste caso é a cor branca (ADOBE SYSTEMS INCORPORATED).

A Figura 38 mostra uma imagem segmentada na qual estão presentes as duas mãos e a cabeça de um indivíduo com os *blobs* destacados de acordo com o algoritmo de busca. No total há três *blobs* na imagem, marcados com cores e enumeração distintas. A interpretação e a forma como as informações fornecidas pela *engine* serão utilizadas é de responsabilidade do ambiente virtual, pois cada ambiente tem a liberdade de utilizar as informações da forma que for conveniente.



Figura 38 – Imagem segmentada com os *blobs* destacados

O ambiente virtual proposto neste trabalho utiliza apenas um *blob* para realizar o controle das ações. Pode haver casos nos quais há a necessidade da representação de mais de um membro no sistema. Portanto, o algoritmo de análise da imagem separa até 3 *blobs*, estabelecendo como critério de seleção a área do retângulo por ordem decrescente.

Caso a imagem segmentada apresente pequenos ruídos, estes elementos podem ser ignorados, pois juntamente com o critério de seleção, a área mínima e máxima do retângulo que delimita os *blobs* é parametrizada. Logo, os *blobs* selecionados são os 3 maiores dentro de uma área estipulada.

Finalmente, a *engine* dispara um evento que contém a lista dos *blobs* classificados. Este evento é disparado toda vez que ocorre todo o processo descrito na Figura 30. Para que não ocorra atraso, esta atividade deve ser executada 15 vezes por segundo.

Conforme descrito anteriormente, o ambiente virtual possui a responsabilidade de efetuar as atualizações que representam o comportamento de um mundo. Por exemplo, o indivíduo deve pegar uma bola que cai em uma determinada velocidade utilizando um marcador posicionado na mão direita. Para que ocorra o efeito de deslocamento, a posição da bola e do cursor que representa a mão deve ser atualizadas a cada vez que o evento é disparado. Esta operação é viabilizada de acordo com a comunicação com os módulos externos que utilizam a *engine*.

7.1.6 Comunicação com Módulos Externos

Após a conclusão da etapa anterior, a *engine* está apta a enviar as informações sobre o quadro analisado para o objeto que estiver aguardando uma notificação de atualização. O envio destas informações não funciona de forma direta; ela é transmitida de forma unidirecional por meio de eventos, que funcionam como se fosse um *broadcast*.

Os objetos são capazes de escutarem o envio dos eventos por meio do método *addEventListener*, que tem como parâmetro o tipo do evento e o método que irá tratá-lo. Conforme explicado anteriormente, um objeto da classe *MotionEvent* é disparado contendo as informações dos *blobs* identificados na imagem em um determinado momento.

O método que trata o evento é o ponto inicial no qual serão efetuadas as ações conforme o estado em que o jogo estará. É neste momento que ocorre o processamento para posicionar os cursores nos locais indicados e realizar a interação destas informações com o ambiente do cenário projetado.

7.2 Resultados Obtidos

Nos itens anteriores, foi apresentado o funcionamento interno da *engine*, que é utilizada para controlar os ambientes virtuais desenvolvidos. Os parâmetros a serem ajustados estão disponíveis para que o usuário possa alterá-los via controles de menu.

O objetivo inicial da *engine* era de realizar a segmentação da pele sem o auxílio de marcadores. A Figura 39 mostra a *engine* sendo executada ainda em fase de desenvolvimento. O treinamento foi executado utilizando a cor da pele. Neste caso, foi identificada a presença de três *blobs*, que representam os braços e a cabeça.

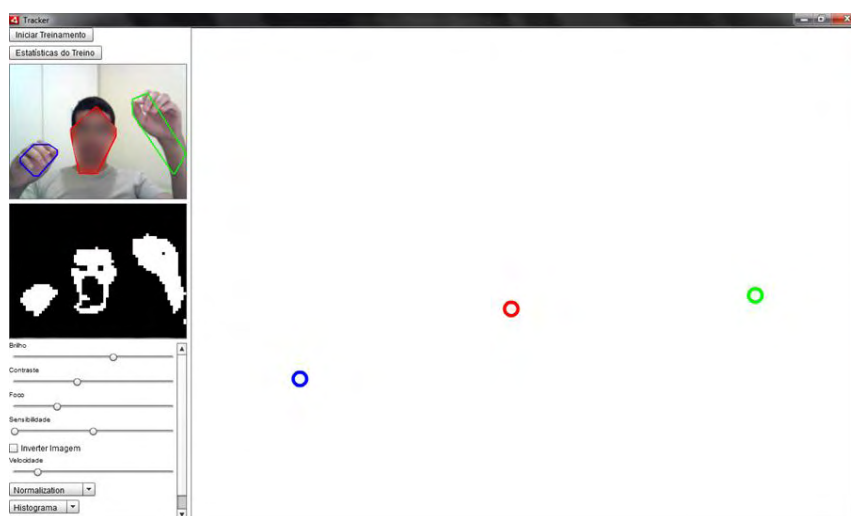


Figura 39 – *Engine* em execução

Os métodos de segmentação avaliados permitem este tipo de abordagem. Entretanto, há alguns fatores que restringem a utilização do sistema pela identificação da pele. Primeiramente, a oclusão entre os membros faz com que a imagem, depois de segmentada, faça a fusão entre os membros, conforme demonstrado na Figura 40.

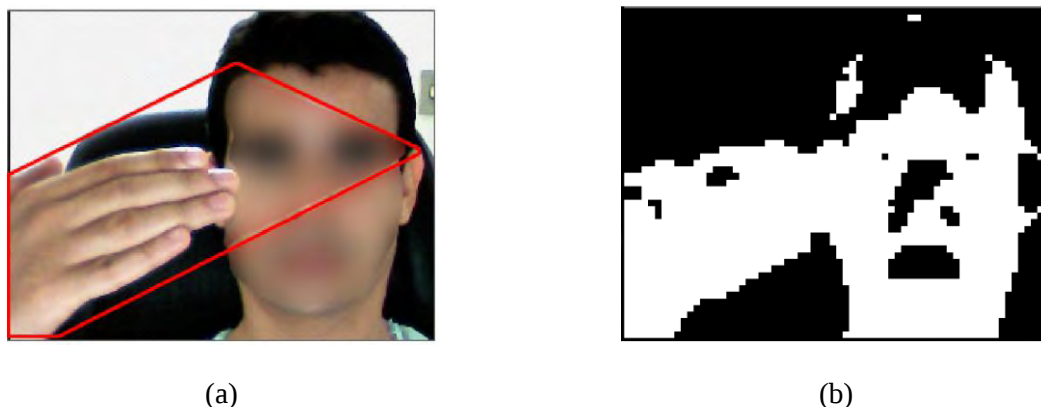


Figura 40 – Oclusão entre os membros. (a) Imagem original; (b) Imagem segmentada

A solução para este tipo de problema exige que a aplicação seja dedicada à extração das características das mãos e do rosto. Poderiam ser utilizadas técnicas de extração de bordas e esqueletonização da imagem (SKARBNIK;SAGIV; ZEEVI, 2009). Há também métodos que trabalham especificamente com o reconhecimento da face utilizando a identificação dos olhos e da boca como pontos de referência (MURPHY-CHUTORIAN; TRIVEDI, 2009).

No entanto, adotar uma abordagem mais específica para o processo de trajetORIZAÇÃO de objetos faz com que a flexibilidade seja perdida, impossibilitando o rastreamento de outros objetos que não sejam os membros superiores (mãos e cabeça) baseados na identificação da pele.

Ao utilizar o tom da pele também há a restrição quanto ao raio de ação que o usuário possui em relação à área de cobertura da *webcam*. Os *blobs* extraídos da imagem ficam muito grandes e ocupam grande parte da área da imagem. Já foram utilizadas luvas coloridas para realizar o mapeamento 3D das mãos no mundo virtual (WANG; POPOVI, 2009), enquanto que outros pesquisadores utilizaram o tom da pele com a restrição de usar o sistema sempre com camisas de manga longa para evitar a oclusão entre os membros (CABRAL;MORIMOTO; ZUFFO, 2005).

Portanto, pelas razões expostas acima, optou-se por abandonar a segmentação de pele e passou-se a utilizar marcadores visuais. A utilização de marcadores visuais possui a vantagem de escolher marcadores de acordo com o tamanho e o formato com o intuito de obter uma boa relação da distância entre o usuário e a câmera

A experiência que o usuário possui ao interagir com a ferramenta é a mesma de um sistema de realidade virtual *desktop*, ou seja, a concentração e os sentidos não estão totalmente destinados ao mundo virtual. Elementos externos podem chamar a atenção e desviar a concentração do usuário do ambiente virtual.

O *feedback* visual é demonstrado na Figura 41 por meio do painel superior, o qual é destinado à visualização dos quadros da imagem capturada e segmentada. O painel inferior contém as configurações dos parâmetros de pré-processamento e outras variáveis que influenciam na utilização do sistema.

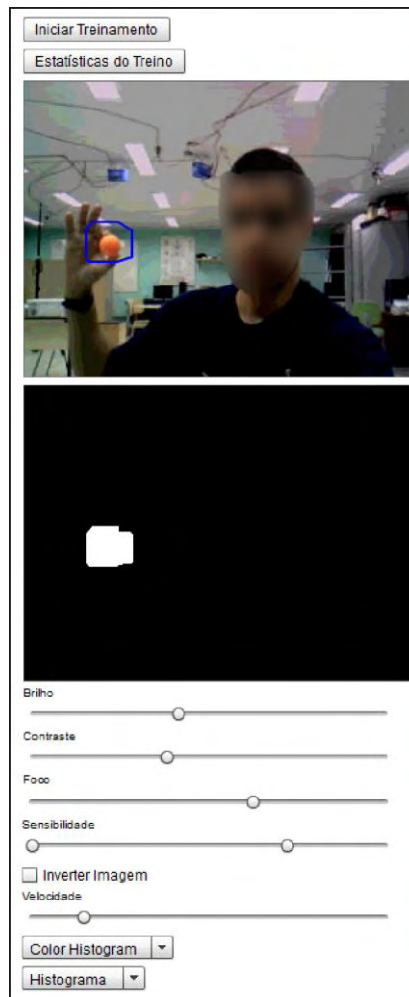


Figura 41 – Painel de configuração da *engine*

O próximo capítulo apresenta o projeto do ambiente virtual que utiliza a *engine* criada neste trabalho. Será mostrada a descrição conceitual e a construção do primeiro ambiente virtual. A modelagem do ambiente virtual foi realizada por meio de visitas técnicas ao CRER, local em que foi possível conhecer com maiores detalhes o ambiente de terapia e as principais necessidades dos pacientes e terapeutas.

8 PROPOSTA DE AMBIENTE VIRTUAL

O ambiente virtual que utiliza a *engine* descrita anteriormente tem como objetivo trabalhar membros superiores de forma que o usuário realize movimentos de alcance em diversas regiões pré-estabelecidas. O modelo baseia-se em uma sequência de alvos distribuídos no plano cartesiano que são posicionados pelo terapeuta.

A monitoração do ambiente virtual se dá pelo uso de objetos que servem como marcadores visuais, os quais o paciente deve segurar com o membro comprometido, de forma a estimular a preensão e a manipulação de objetos.

O paciente deve concentrar seus esforços para que o alcance dos pontos seja realizado de forma eficiente, ou seja, deve evitar a ocorrência de compensação na postura e definir melhores estratégias motoras para a realização da tarefa.

8.1 Modelo Conceitual

O ambiente virtual projetado tem como objetivo estabelecer alvos para que o usuário faça o movimento de alcance até cada um deles. O terapeuta será capaz de definir a quantidade e a posição de cada alvo com o intuito de adequar a tarefa para as limitações motoras de cada paciente.

Durante as visitas técnicas realizadas no CRER, foi observada a terapia empregada em pacientes com hemiplegia oriunda de AVE. A intervenção realizada pelos profissionais de terapia ocupacional e fisioterapia consiste em estimular a neuroplasticidade de forma que os pacientes consigam exercer o controle motor dos membros comprometidos.

O exercício, no qual foi baseada a modelagem do ambiente virtual, consiste em posicionar o paciente em pé na frente de um espelho, enquanto o terapeuta ocupacional fixa marcadores adesivos no próprio espelho. De acordo com as orientações do terapeuta, o paciente deve realizar movimentos com o lado hemiplégico de tal forma que os marcadores sejam tocados.

O esquema da Figura 42 demonstra como é realizado o exercício em frente ao espelho. A dificuldade do exercício pode ser configurada de acordo com a distância entre o paciente e o espelho. Os pacientes apresentam sinergia (a contração de músculos agonistas e antagonistas ao mesmo tempo) e espasticidade (dificuldade de movimentar uma articulação de forma passiva) nos membros afetados pela lesão (PIZZI *et al.*, 2005). Conforme o posicionamento dos alvos, o terapeuta consegue mensurar a restrição aos movimentos provocada pela espasticidade e sinergia. De maneira geral, os alvos que estão acima do ombro e no lado contralateral são mais difíceis de serem alcançados, pois exigem uma estratégia

motora mais complexa devido ao movimento simultâneo da articulação do ombro e do cotovelo.

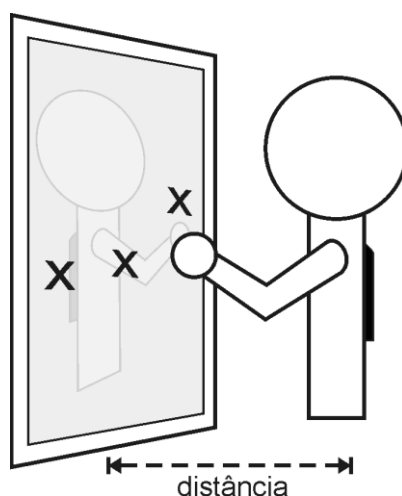


Figura 42 – Esquema do exercício realizado em frente ao espelho

O ambiente virtual desenvolvido trabalha sob a mesma perspectiva do exercício praticado em frente ao espelho. Porém, ao invés de reproduzir a experiência do espelho de forma fiel, o mundo virtual possui a representação de um parque no qual o paciente deve estourar balões com alfinetes que são manipulados pelos movimentos das mãos. A Figura 43 apresenta o esboço inicial do ambiente virtual.

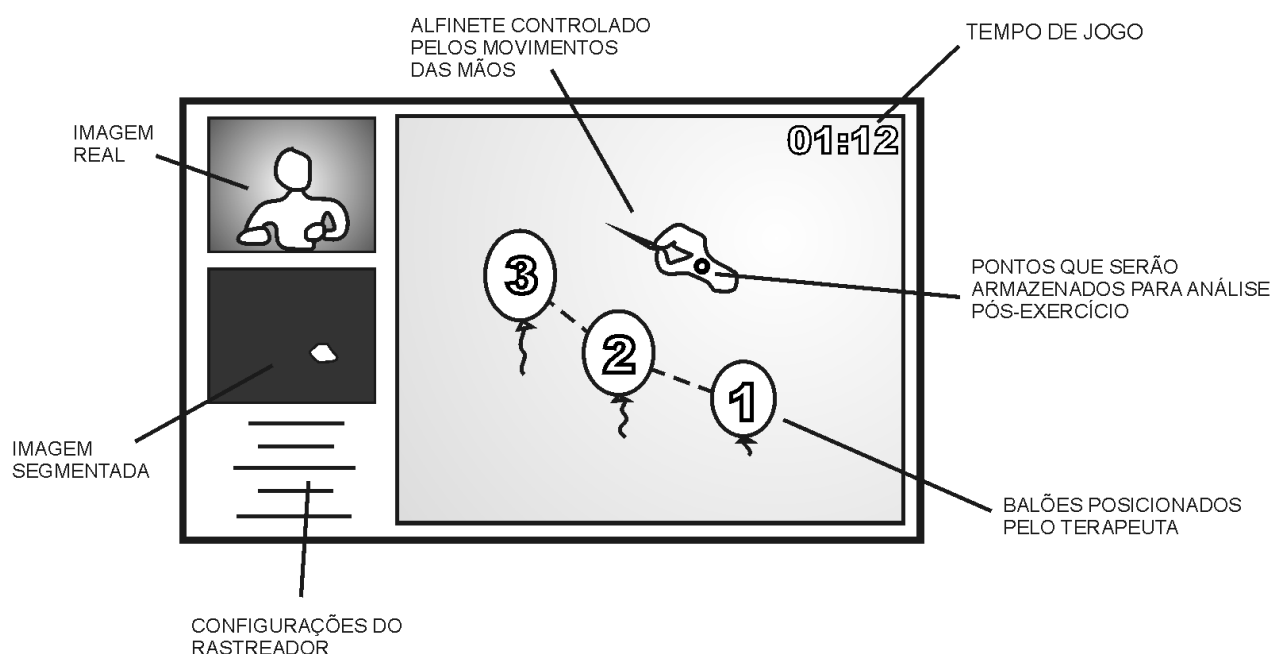


Figura 43 – Esboço do ambiente virtual

De acordo com o *layout* da tela, a área à esquerda é destinada à visualização das imagens reais e segmentada juntamente com os controles de configurações da *engine*. O restante da tela é destinado ao próprio ambiente virtual. O ambiente virtual contém a

ilustração do alfinete que é manipulado pelo usuário e os balões, que serão enumerados de acordo com a ordem em que deverão ser estourados.

As ilustrações do cenário, alfinete e balões foram elaboradas para que o usuário não tenha distrações durante a realização da atividade. O cenário do protótipo não contém animações e as cores contrastam com os elementos que são de interação (balão e alfinete) no ambiente virtual. A Figura 44 mostra as ilustrações dos elementos gráficos que fazem parte do ambiente virtual. Nota-se a presença da agulha, do balão e do plano de fundo no qual os objetos são projetados.

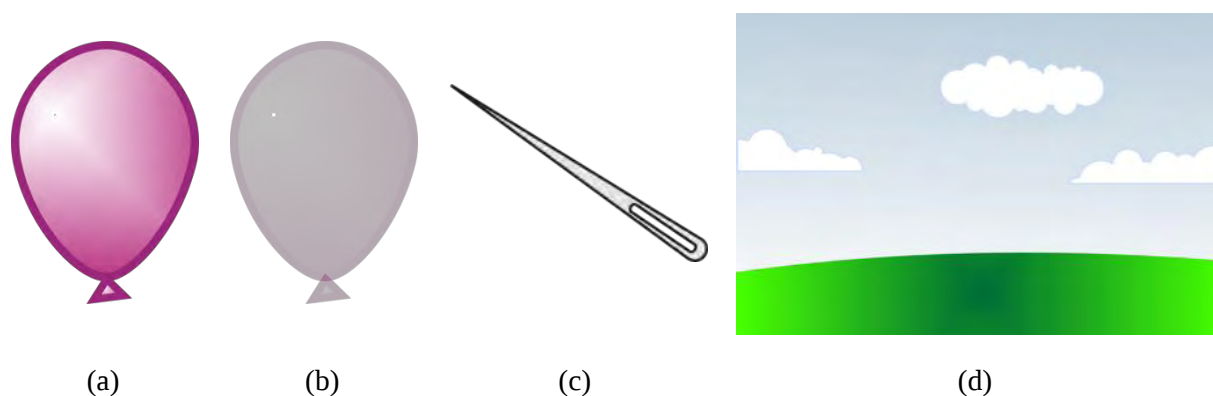


Figura 44 – Elementos gráficos do ambiente virtual. (a) Balão; (b) Balão quando é estourado; (c) Alfinete; (d) Plano de fundo

A utilização do ambiente virtual é bem simples. Não há recursos de navegação avançados; o usuário deve basicamente segurar um marcador para que a agulha seja manipulada.

Os indivíduos que farão uso do ambiente virtual são, em sua maioria, pessoas que utilizam o Sistema Único de Saúde (SUS). Os pacientes possuem diferentes níveis de experiência em jogos eletrônicos e na maioria dos casos são pessoas idosas. Portanto, a curva de aprendizado para o domínio do ambiente virtual é bastante simplificada.

Diferentemente do exercício realizado em frente ao espelho, o terapeuta pode mensurar a produtividade do paciente durante o uso do ambiente virtual. Há duas variáveis que podem ser obtidas pela ferramenta a fim de gerar informações relacionadas ao desempenho da atividade executada pelo paciente: o tempo e a posição espacial do marcador em cada instante do exercício.

8.2 Informações de Saída

O terapeuta deve trabalhar com o paciente buscando metas e objetivos a serem alcançados. Um recurso que ajuda o terapeuta é a mensuração da evolução do paciente

durante determinado exercício, uma vez que “só pode ser gerenciado o que pode ser medido” (KAPLAN; NORTON, 1997).

Existem protocolos pré-estabelecidos para a avaliação do controle motor de membros superiores como, por exemplo, Análise Funcional Escolar (AFE) e Diário de Atividade Motora (DAM) (SHUMWAY-COOK; WOOLLACOTT, 2007). Esses protocolos avaliam as atividades motoras de acordo com a aplicação de questionários nos quais o terapeuta estabelece pontuações de acordo com a capacidade que o paciente tem em realizar diversas atividades cotidianas.

Contudo, esses protocolos de avaliação citados estabelecem uma pontuação sobre a evolução do paciente de forma geral, independente do tipo de intervenção adotada. O foco das informações de saída do ambiente virtual é de verificar como o paciente está avançando nesse tipo de exercício. Este indicador de desempenho não possui o objetivo de substituir os métodos de avaliação clássicos, mas completá-los. Vale ressaltar que, durante a terapia, o paciente não realiza apenas um tipo de tarefa; o resultado é obtido de acordo com o conjunto de atividades executadas durante o processo de recuperação funcional.

Durante a execução do exercício, o ambiente virtual armazena as coordenadas da trajetória realizada pelo paciente e o tempo gasto para a conclusão da tarefa. Estas variáveis fornecem informações relativas ao progresso do jogador sobre o uso do ambiente virtual.

Ao concluir a atividade, o ambiente virtual apresenta dois tipos de gráficos diferentes: um gráfico que representa a posição dos balões e a trajetória realizada e outro, que compara a trajetória programada com a trajetória realizada distribuídas no tempo.

O primeiro gráfico utiliza a posição x e y dos balões e do objeto rastreado em pixels. A Figura 45 mostra a saída de uma partida na qual o ambiente virtual foi configurado com sete balões. Os círculos representam a posição dos balões enquanto a linha é a trajetória realizada em uma partida qualquer.

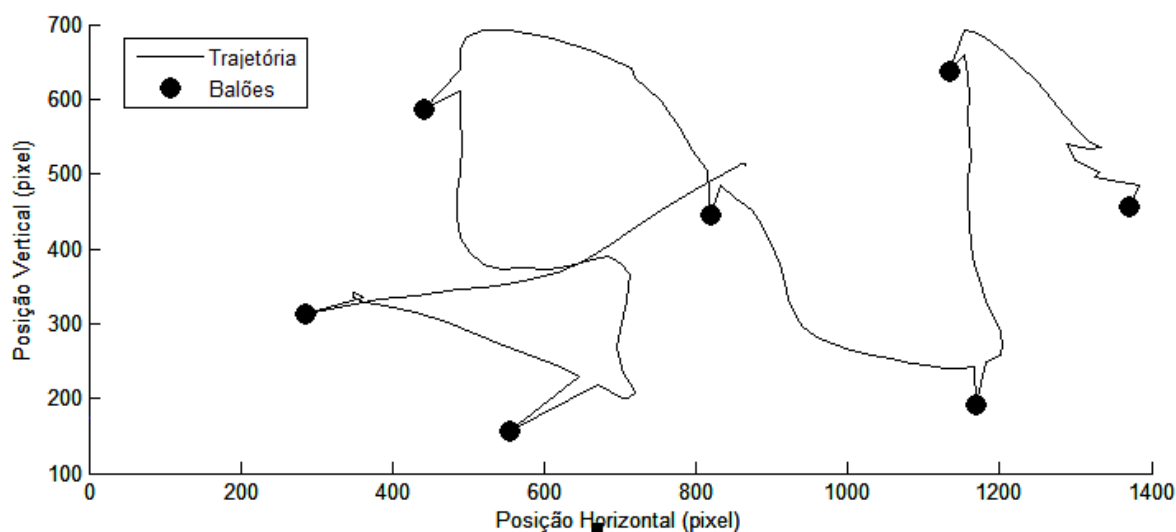


Figura 45 – Gráfico da trajetória realizada

No segundo gráfico, a trajetória programada consiste em retas que interligam os balões enquanto a trajetória realizada mostra exatamente a movimentação do usuário até o final da atividade.

O eixo das abscissas representa os instantes em que as coordenadas dos objetos rastreados foram capturadas. Já o eixo das ordenadas representa a coordenada y do objeto rastreado em pixel. Inicialmente, a série programada não possui a informação temporal, ela é um conjunto de pontos que representam a posição dos balões.

Para distribuir a série programada no tempo, foi utilizada a série realizada como referência para saber o instante em que os pontos são alcançados. É nesse instante que o ponto programado é inserido no gráfico e a interpolação entre um ponto e outro é realizada por meio de retas que interligam os pontos programados.

Com as informações das séries programada e realizada, é calculada uma nova série que representa o erro. A série erro é o módulo da diferença entre as séries programada e realizada. A série erro permite avaliar a precisão que o paciente teve durante a execução do exercício.

A Figura 46 mostra o gráfico que o ambiente virtual apresenta para o terapeuta. O gráfico possui a série para a trajetória programada, a trajetória realizada pelo paciente e a série erro. Os instantes em que o erro se iguala a zero é o momento em que o alvo é alcançado. A análise do gráfico permite verificar em quais pontos o paciente tem maior dificuldade motora para realizar a tarefa.

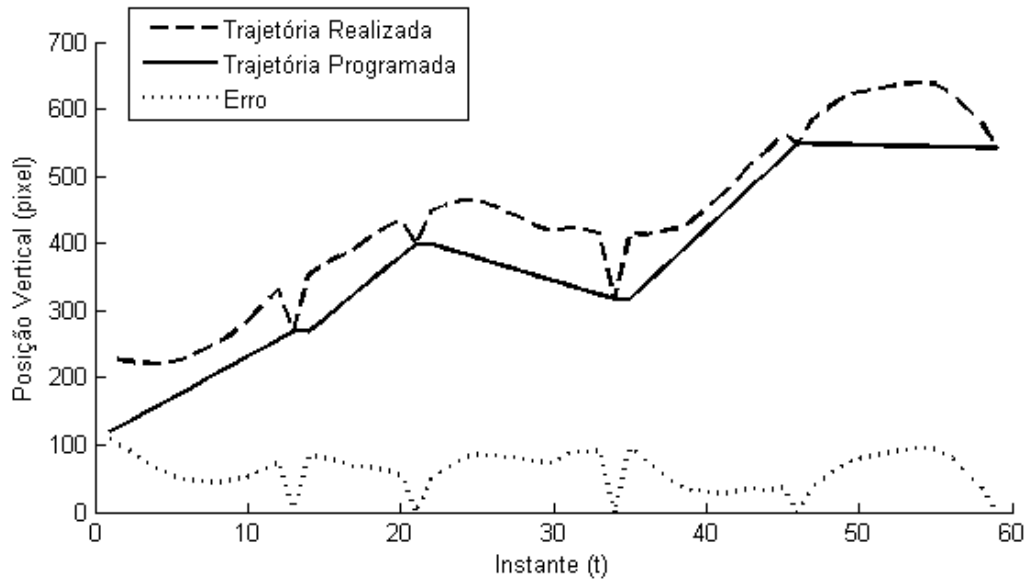


Figura 46 – Trajetória realizada em relação à trajetória programada distribuídas no tempo

A partir das informações da série erro é calculado um valor que representa a pontuação da precisão de forma global. Foi utilizado o Erro Quadrático Médio (EQM) para calcular a nota que representa a precisão do paciente, pois o EQM não depende de parâmetros e nem de informações adicionais para ser calculado e estabelece uma relação euclidiana entre a distância do ponto ideal (quando o erro é nulo) e o cenário calculado (ZHOU; BOVIK, 2009).

A definição do EQM é definida pela equação (13):

$$EQM = \frac{1}{n} * \sum_{i=1}^n (e_i)^2 \quad (13)$$

onde n é o número de amostras da série e e é o valor do erro.

Além da precisão, o tempo gasto para a realização da tarefa também é utilizado como informação para o terapeuta. Levando em conta que a tarefa é dada como concluída quando uma sequência de alvos é alcançada, o tempo de um alvo para o outro pode ser armazenado. Esta ideia permite que o terapeuta identifique se as variações de posição entre os balões afetam o desempenho do paciente.

O desempenho global da tarefa depende tanto da precisão quanto do tempo. Logo, uma tarefa é considerada bem executada quanto menos tempo for gasto na atividade e também quanto menor for o EQM. Pode ser que os dois fatores não sejam fáceis de serem melhorados em um mesmo momento da terapia. Cabe ao terapeuta saber se quer melhorar o tempo ou a precisão dos movimentos do paciente.

8.3 Arquitetura

Foi adotado o mesmo paradigma de programação do desenvolvimento da *engine* no projeto do ambiente virtual. Com o uso de orientação a objetos, foram abstraídos os principais elementos do cenário do ambiente virtual de acordo com a realidade. Pensando no cenário para o mundo real, teríamos que posicionar balões na superfície de tal forma que ele ficasse estático. No momento que o exercício fosse iniciado, uma pessoa iria estourá-los um de cada vez utilizando um alfinete.

O diagrama de classes da Figura 47 representa a forma como foi modelada a abstração do cenário real para o ambiente de programação. As classes com a cor vermelha fazem parte do *kit* de desenvolvimento do Flex, enquanto as classes de cor branca foram criadas sob o contexto do ambiente virtual.

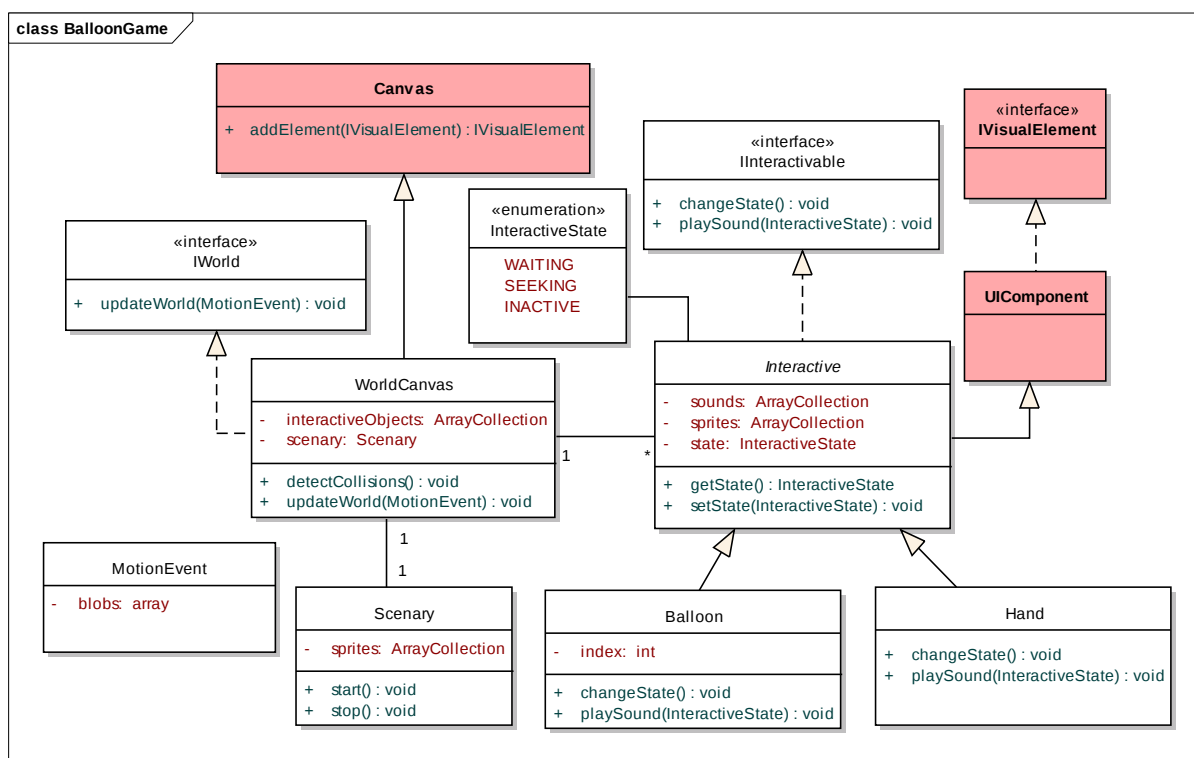


Figura 47 – Diagrama de classes do ambiente virtual

Em primeiro lugar, o mundo virtual é representado pela classe *WorldCanvas*, na qual os demais elementos do ambiente virtual serão inseridos. É nesta classe que as características físicas e de atualizações de tela são processadas. O método *updateWorld* é responsável por receber o evento disparado pela *engine* e, por meio das informações contidas no evento, atualizar a posição dos marcadores visuais. Já o método *detectCollisions* verifica se há colisões entre os objetos estáticos e aqueles que se movimentam na tela.

A classe *Scenary* possui os elementos de plano de fundo e algumas imagens que dão forma ao ambiente virtual. Depois que os balões são posicionados na tela, o terapeuta inicia a partida por meio do método *start*. Quando o paciente estoura todos os balões ou o tempo de partida se esgota, o método *stop* é disparado e a partida é encerrada.

Os objetos que apresentam algum tipo de interação são representados primeiramente pela interface *IInteractivable* realizada pela classe abstrata *Interactive*, a qual possui algumas informações básicas como o estado atual, imagens utilizadas na renderização (*sprites*) e os sons que são disparados nas mudanças de estado.

Os herdeiros da classe *Interactive* são as classes *Balloon*, que representam os alvos ou balões e a classe *Hand* que representa a mão que carrega o alfinete. A interação entre essas duas classes são controladas por meio de diagrama de estados. Os eventos de início de jogo e o de colisão causam a alteração destes estados. A Figura 48 apresenta o diagrama de máquina de estados das classes que herdam *Interactive*.

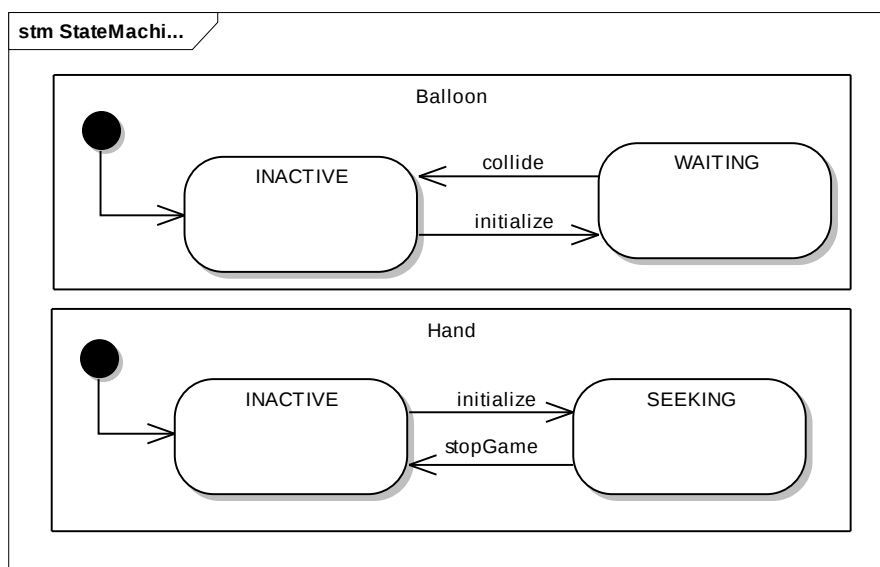


Figura 48 – Máquina de estados das classes herdeiras de *Interactive*

Os balões são inseridos antes de começar uma partida. Eles são inicializados com o estado *INACTIVE*. Quando uma partida é iniciada, os balões passam do estado *INACTIVE* para o estado *WAITING*. O alfinete é inicializado da mesma forma com o estado *INACTIVE*. Ao iniciar a partida, ele altera para o estado *SEEKING*, ou seja, o alfinete está procurando um balão para poder estourá-lo. Quando é detectada uma colisão entre um objeto da classe *Hand* no estado *SEEKING* com um objeto da classe *Balloon* no estado *WAITING*, entende-se que o balão seja estourado neste momento, logo, seu estado é alterado para *INACTIVE*.

Quando há a alteração de estados destes objetos, o método *changeState* é executado para que ocorra alteração visual ou até mesmo animações. Os efeitos sonoros dos objetos de interação também são disparados de acordo com o novo estado a ser utilizado pelo método *playSound*.

Portanto, o ambiente virtual foi desenvolvido de tal forma que a *engine* fique completamente desacoplada dele. O único meio de comunicação entre eles é realizado de forma unidirecional com o evento *MotionEvent*. O desenvolvimento de novos ambientes virtuais pode ser realizado como forma de *plugin*, em que a comunicação entre *engine* e ambiente virtual seguiria a mesma linha de raciocínio deste primeiro protótipo.

8.4 Resultados e Discussões

Pelo fato do ambiente virtual ser do tipo *desktop*, não é de se esperar que o usuário tenha total imersão. Porém, há alguns fatores que contribuem para um bom aproveitamento da atividade executada. A partir dos testes relatados nesta seção, foi observado que o tamanho do monitor torna a atividade mais prazerosa e o usuário fica menos suscetível a distrações. A acústica do lugar de utilização do ambiente virtual também é importante para manter a concentração do usuário. Há pessoas que se sentem mais à vontade e conseguem se concentrar melhor em locais silenciosos (SECOLI *et al.*, 2011).

Os sentidos que são estimulados são a visão, a audição e o tato, este último por meio da influência do marcador visual. O marcador visual pode ser qualquer objeto. Logo, o terapeuta poderá utilizar um dado objeto pelo seu tamanho, formato, cor e massa conforme a necessidade de cada paciente.

O ambiente virtual foi testando em diferentes locais e com monitores de 15, 22, 27 e 32 polegadas. A iluminação dos locais de testes era artificial com lâmpadas fluorescentes fixadas no teto. O sistema foi testado por pessoas vinculadas ao LabioEng tanto sentado como em pé, havendo apenas o ajuste da inclinação da câmera para que o enquadramento ficasse na área de movimentação.

A Figura 49 mostra o esquema do teste realizado no monitor de 32 polegadas, no qual a resolução da tela estava configurada em 1920 por 1080 pixels. A pessoa ficou posicionada a uma distância de 190 centímetros da *webcam* e o marcador visual utilizado foi um disco da cor verde de 20 centímetros de diâmetro.

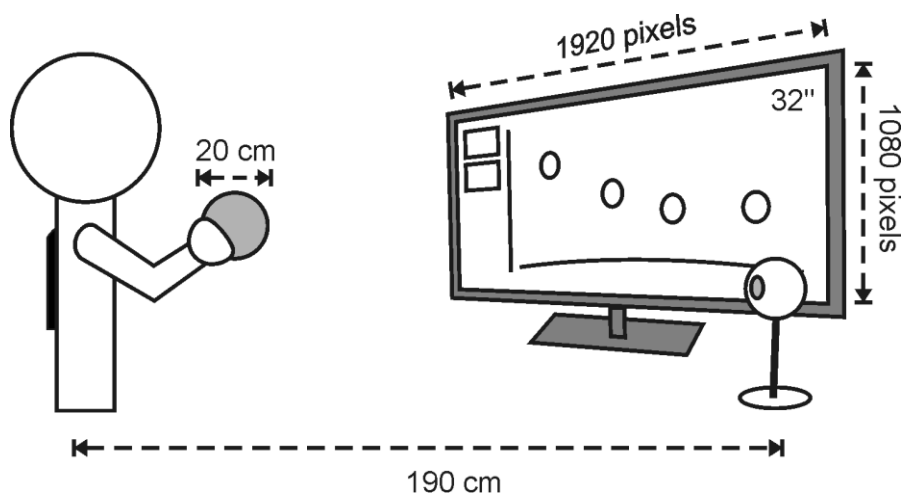


Figura 49 – Esquema da disposição dos objetos durante o uso do ambiente virtual

Na Figura 50 é apresentada a captura da tela no momento em que o ambiente virtual estava sendo utilizado com a configuração citada. Os balões podem ser distribuídos a uma distância maior ou menor entre eles conforme o tamanho da área da tela destinada para o ambiente virtual.

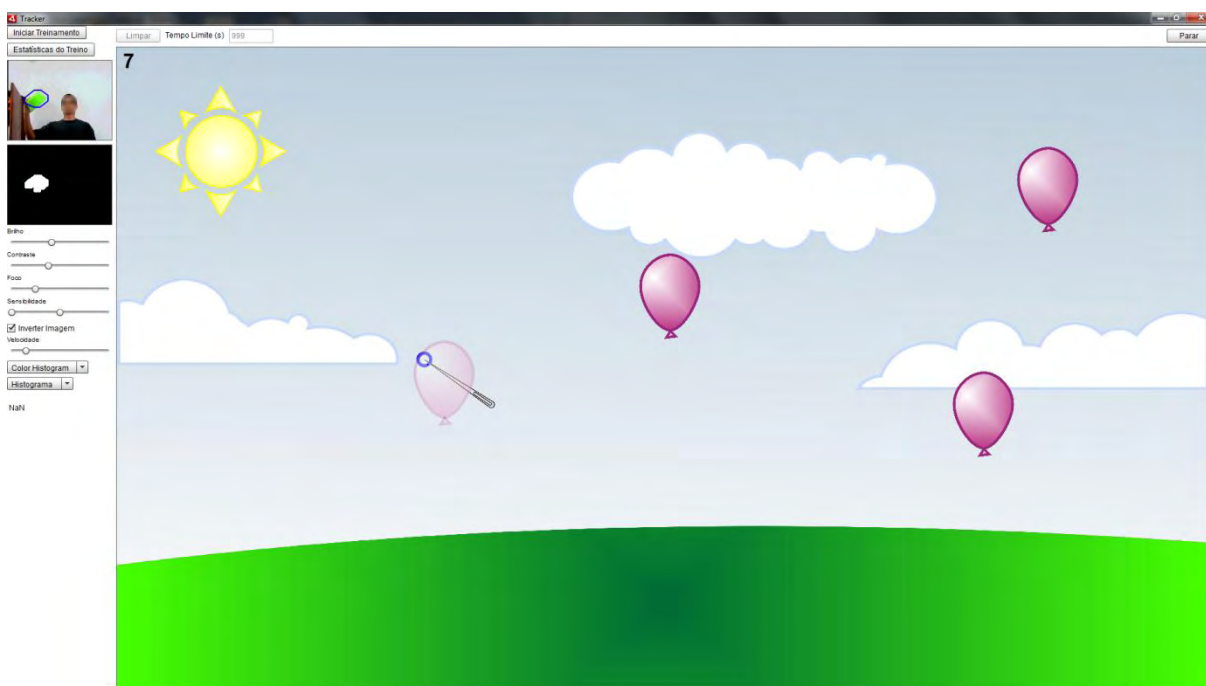


Figura 50 – Ambiente virtual na resolução de 1920 por 1080 pixels

A amplitude do movimento que o usuário pode realizar depende diretamente do enquadramento da *webcam* e da distância que a *webcam* está posicionada em relação a ele. Deve-se levar em consideração também o tamanho do marcador visual a ser utilizado; o seu tamanho é diretamente proporcional à distância entre o usuário e a *webcam*.

Comparando com o mundo real, o ambiente virtual não fornece a profundidade do objeto rastreado. O usuário deve alcançar o alvo dentro do plano bidimensional, enquanto no mundo real as atividades são executadas no plano tridimensional. Porém, para que uma pessoa realize a tarefa, a combinação do movimento dos músculos e das articulações faz com que o movimento realizado pelo usuário esteja em uma perspectiva tridimensional.

Espera-se que a utilização do dispositivo em ambiente de terapia contribua para que as expectativas do terapeuta em relação ao progresso do paciente sejam alcançadas. Para o paciente o dispositivo será visto como uma novidade em relação às atividades corriqueiras. Os estímulos audiovisuais que o dispositivo produz faz com que o paciente tenha maior interesse em realizar as tarefas pré-determinadas.

9 CONCLUSÃO

O processo de criação do ambiente virtual e do dispositivo de rastreamento permitiu o alcance do objetivo principal deste trabalho. Conforme as visitas realizadas no CRER, constatou-se que o investimento em tecnologia assistiva traz benefícios, tanto para o paciente em terapia quanto para o profissional da saúde.

Os produtos que atendem este nicho, tais como softwares, robôs e dispositivos mecânicos que auxiliam a terapia possuem custo bastante elevado. A acessibilidade que um paciente possui sobre um determinado produto está apenas em ambiente clínico. De acordo com alguns relatos e quadros clínicos analisados, há pacientes que não completam todo o processo de terapia por diversos motivos, sendo os principais: a dificuldade de locomover-se às clínicas e a falta de motivação causada pelo longo período de terapia.

O tipo de dispositivo desenvolvido permite que o paciente, quando habilitado pelo terapeuta, possa realizar alguns exercícios fora do ambiente clínico e enviar o resultado das atividades para os terapeutas. O custo que o paciente terá para conseguir utilizar a ferramenta é de um computador pessoal e de uma *webcam*.

Todavia, cabe ao terapeuta avaliar cada paciente para identificar aqueles que estão disciplinados ao ponto de realizar as atividades sem a presença de um profissional da saúde. Este estudo não tem o objetivo de substituir a intervenção utilizada nas clínicas, é apenas uma ferramenta extra que o terapeuta possui dentro de um leque de opções.

Ainda não se pode afirmar que o uso do ambiente virtual desenvolvido neste trabalho traga benefícios para os pacientes maiores que a terapia convencional. Seria necessário realizar o trabalho de validação em clínicas de reabilitação com um grupo utilizando o ambiente virtual e um grupo utilizando a terapia convencional. Entretanto, podemos adiantar que pacientes aos quais foi mostrado o sistema, mostraram-se bastante estimulados pela novidade da ferramenta que traz cores, formas e sons para as sessões de terapia.

Do ponto de vista tecnológico, a *engine* do ambiente virtual permite que novos ambientes sejam desenvolvidos, inclusive para outras partes do corpo que não necessariamente membros superiores. Os marcadores visuais podem ser fixados em qualquer parte do corpo, possibilitando a criação de novos ambientes virtuais que explorem outros movimentos de outros membros, como por exemplo, o movimento de extensão de perna.

9.1 Considerações Finais

Este projeto abre a possibilidade de dois tipos de expansão. Primeiramente do lado tecnológico adicionando novos dispositivos que fazem parte do rastreamento dos objetos de interesse e também a criação de novos ambientes virtuais por meio de parcerias com pesquisadores da área da saúde.

Durante a execução deste projeto surgiram dois sensores de movimento no mercado que podem ser úteis para o enriquecimento deste trabalho. Lançados inicialmente para o mercado de videogames, tanto o PlayStation® Move da Sony™ quanto o Kinect da Microsoft™ trabalham com o objetivo de utilizar a visão computacional como forma de fornecer ações de entrada para os jogos.

O PlayStation® Move assemelha-se ao que foi criado nesse trabalho. O mecanismo consiste em um *joystick* cilíndrico que possui uma esfera com iluminação própria em sua extremidade. Uma câmera é ligada ao console que processa as imagens adquiridas por meio dela. A esfera possui a particularidade de ajustar a cor conforme o ambiente em que esteja sendo utilizado.

O sensor de movimento Kinect possui um conjunto de duas câmeras e um feixe de luz infravermelha de tal forma que os algoritmos de rastreamento sejam capazes de reconstruir o esqueleto humano dentro do espaço tridimensional.

Uma próxima etapa para este estudo seria estudar o kit de desenvolvimento destes dois sensores de movimento para saber quais são as limitações de cada um deles. Após efetuar a escolha dos sensores, a *engine* seria reprogramada para que aceitasse também o uso de um desses sensores.

Em uma avaliação superficial, percebe-se que a utilização do Kinect seria mais apropriada para aplicações assistivas, uma vez que o dispositivo foi desenvolvido para trabalhar com o modelo do corpo humano. Poderiam ser desenvolvidos ambientes virtuais que exigissem mais complexidade do ponto de vista de rastreamento ao combinar vários movimentos do corpo humano, deixando de lado a questão de utilização de marcadores visuais.

Conforme citado anteriormente, este projeto também abre caminho para o desenvolvimento de um conjunto de aplicativos destinados ao ambiente de terapia, podendo trabalhar tanto o controle motor como os aspectos cognitivos de um paciente.

Essas duas propostas de trabalho podem ser realizadas paralelamente, uma vez que a modelagem de novos ambientes virtuais é completamente independente da tecnologia

de entrada a ser adotada, ou seja, a tecnologia a ser empregada deve suprir os requisitos do ambiente virtual modelado.

REFERÊNCIAS

- ADKINS, D. L. *et al.* Motor training induces experience-specific patterns of plasticity across motor cortex and spinal cord. **J Appl Physiol**, v. 101, n. 6, p. 1776-82, Dec 2006.
- ADOBE SYSTEMS INCORPORATED. ActionScript® 3.0 Reference for the Adobe® Flash® Platform. Disponível em: <http://help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/flash/display/BitmapData.html>. Acesso em: 10/06/2011.
- ANDRADE, A. L. M.; JUNIOR, A. L. A plasticidade neural e suas implicações nos processos de memória e aprendizagem. **RUBS**, v. 1, n. 3, p. 27-32, 2005.
- ARVO, J. **Graphics Gems II**. Academic Press, 1991.
- ASSIS, G. A. D. *et al.* Neuror: Sistema de Realidade Aumentada para Reabilitação Física de Pacientes Vítimas de Acidente Vascular Encefálico. In: Congresso Brasileiro de Informática na Saúde, Campos do Jordão - São Paulo. 2008.
- BLANCHARD, C. *et al.* Reality built for two: a virtual reality tool. **SIGGRAPH Comput. Graph.**, v. 24, n. 2, p. 35-36, 1990.
- BLEIWEISS, A. *et al.* Enhanced interactive gaming by blending full-body tracking and gesture animation. *ACM SIGGRAPH ASIA 2010 Sketches*. Seoul, Republic of Korea: ACM, 2010. p. 1-2.
- BOTELLA, C. *et al.* Treating cockroach phobia using a serious game on a mobile phone and augmented reality exposure: A single case study. **Computers in Human Behavior**, v. 27, n. 1, p. 217-227, 2011.
- BRAND, J.; MASON, J. S. A comparative assessment of three approaches to pixel-level human skin-detection. In: Pattern Recognition, 2000. Proceedings. 15th International Conference on, 2000. 2000. p.1056-1059 vol.1.
- BYL, N. *et al.* Effectiveness of sensory and motor rehabilitation of the upper limb following the principles of neuroplasticity: patients stable poststroke. **Neurorehabil Neural Repair**, v. 17, n. 3, p. 176-91, Sep 2003.
- CABRAL, M. C.; MORIMOTO, C. H.; ZUFFO, M. K. On the usability of gesture interfaces in virtual reality environments. *Proceedings of the 2005 Latin American conference on Human-computer interaction*. Cuernavaca, Mexico: ACM, 2005. p. 100-108.
- CHEUNG, S. S.; KAMATH, C. Robust techniques for background subtraction in urban traffic video. in **Proc. SPIE Video Communications and Image Processing**, v., 2004.
- COOK, A. M.; HUSSEY, S. M. **Assistive technologies: principles and practice**. Mosby, 2002.
- CORPORAL, G. Hemiplegia. Disponível em: <[http://www.poderdasmaos.com/site/pub/bancoimg/bancodeimagens/Enciclopedia_Medica/hdc_0001_0003_0_img0192\[1\].jpg](http://www.poderdasmaos.com/site/pub/bancoimg/bancodeimagens/Enciclopedia_Medica/hdc_0001_0003_0_img0192[1].jpg)>. Acesso em: 20/12/2011.

COWLEY, B. *et al.* Learning principles and interaction design for "Green My Place": A massively multiplayer serious game. **Entertainment Computing**, v. 2, p. 103-113, 2011.

DOUGHERTY, E. R.; LOTUFO, R. A. **Hands-on morphological image processing**. SPIE Press, 2003.

ERTELT, D. *et al.* Action observation has a positive impact on rehabilitation of motor deficits after stroke. **Neuroimage**, v. 36 Suppl 2, p. 164-73, 2007.

FISHER, B. E.; SULLIVAN, K. J. Activity-dependent factors affecting poststroke functional outcomes. **Top Stroke Rehabil**, v. 8, n. 3, p. 31-44, Autumn 2001.

FRANCHI, J. Virtual reality: An overview. **TechTrends**, v. 39, n. 1, p. 23-26, 1994.

FREITAS, R. F. *et al.* Algoritmos para segmentação da pele utilizando modelos de cores RGB em ambiente Matlab/Simulink. **Conexões – Ciência e Tecnologia**, v. 1, p. 65-71, 2007.

FUNK, J. B. Reevaluating the Impact of Video Games. **Clinical Pediatrics**, v. 32, n. 2, p. 86-90, February 1, 1993 1993.

GARCIA, R. F. *et al.* Um arcabouço para localização de enxames de robôs. **VIII Simpósio Brasileiro de Automação Inteligente**, v., 2007.

GONZÁLEZ, R. C.; WOODS, R. E. **Digital image processing**. Pearson/Prentice Hall, 2008.

GOUVEIA, L. Introdução aos conceitos de Realidade Virtual. Disponível em: <http://www2.ufp.pt/~lmbg/lg_textos.htm>. Acesso em: 12/12/2011.

GRIBBON, K. T.; BAILEY, D. G. A novel approach to real-time bilinear interpolation. In: Electronic Design, Test and Applications, 2004. DELTA 2004. Second IEEE International Workshop on, 28-30 Jan. 2004. 2004. p.126-131.

HALTON, J. Virtual rehabilitation with video games: A new frontier for occupational therapy. **Occupational Therapy Now**, v. 10, n. 1, p. 12-14, 2008.

HAND, C. Other faces of virtual reality
Multimedia, Hypermedia, and Virtual Reality Models, Systems, and Applications. In: BRUSILOVSKY, P.; KOMMERS, P.; STREITZ, N. (Ed.). Springer Berlin / Heidelberg, 1996. p. 107-116. (Lecture Notes in Computer Science).

HECKBERT, P. Color image quantization for frame buffer display. **SIGGRAPH Comput. Graph.**, v. 16, n. 3, p. 297-307, 1982.

HEILIG, M. Beginnings: Sensorama and the Telesphere Mask *Digital illusion*: ACM Press/Addison-Wesley Publishing Co., 1998. p. 343-351.

HELD, J. M. Environmental Enrichment Enhances Sparing and Recovery of Function Following Brain Damage. **Journal of Neurologic Physical Therapy**, v. 22, n. 2, p. 74-78, 1998.

INVENTORVR. Sensorama Machine. Disponível em: <http://www.mortonheilig.com/InventorVR.html>. Acesso em: 01/02/2012.

ISO. ISO 9999/2002. Disponível em: http://www.iso.org/iso/catalogue_detail.htm?csnumber=33150. Acesso em: 21/12/2011.

JOO, L. Y. *et al.* A feasibility study using interactive commercial off-the-shelf computer gaming in upper limb rehabilitation in patients after stroke. **Journal of Rehabilitation Medicine**, v. 42, n. 5, p. 437-441, 2010.

KAMEL, J. A. N. **Engenharia do Entretenimento: Meu vício, Minha virtude**. Editora E-papers.

KANADE, T.; NARAYANAN, P. J.; RANDE, P. W. Virtualized reality: concepts and early results. In: Representation of Visual Scenes, 1995. (In Conjunction with ICCV'95), Proceedings IEEE Workshop on, 24 Jun 1995. 1995. p.69-76.

KAPLAN, R.; NORTON, D. Why does business need a balanced scorecard? **Journal of Cost Management**, v. 11, n. 3, 1997.

KEYS, R. Cubic convolution interpolation for digital image processing. **Acoustics, Speech and Signal Processing, IEEE Transactions on**, v. 29, n. 6, p. 1153-1160, 1981.

KOSCHAN, A.; ABIDI, M. A. **Digital color image processing**. Wiley-Interscience, 2008.

LANDERS, M. Treatment-induced neuroplasticity following focal injury to the motor cortex. **Int J Rehabil Res**, v. 27, n. 1, p. 1-5, Mar 2004.

LASKO-HARVILL, A. *et al.* From DataGlove to DataSuit. In: Compcon Spring '88. Thirty-Third IEEE Computer Society International Conference, Digest of Papers, 29 Feb-3 Mar 1988. 1988. p.536-538.

LEHMANN, T. M.; GONNER, C.; SPITZER, K. Survey: interpolation methods in medical image processing. **Medical Imaging, IEEE Transactions on**, v. 18, n. 11, p. 1049-1075, 1999.

LENT, R. **Cem Bilhões de Neurônios: conceitos fundamentais de neurociências**. São Paulo: Atheneu, 2004.

LUNDY-EKMAN, L. **Neurociência: Fundamentos Para a Reabilitação**. 3. ed. São Paulo: Elsevier (Medicina), 2008.

MATHWORKS. RGB Color Cube for uint8 Images. Disponível em: <http://www.mathworks.com/help/toolbox/images/color7.gif>. Acesso em: 30/11/2011.

MAURI, C. *et al.* Computer Vision Interaction for People With Severe Movement Restrictions. **An Interdisciplinary Journal on Humans in ICT Environments**, v. 2, n. 1, p. 38-54, 2006.

MAZALEK, A. *et al.* I'm in the game: embodied puppet interface improves avatar control. **Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction**, v. 5, p. 129-136, 2011.

MICROSOFT. Microsoft Kinect. Disponível em: <<http://www.xbox.com/pt-BR/Xbox360/Accessories/Kinect/>>. Acesso em: 15/02/2012.

MINELLI, C. Mecanismos da Neuroplasticidade Pós-AVC. *Boletim Neuro Atual*, v. 2. Disponível em: <<http://familiabrasil.org/revista/ojs-2.2.3/index.php/ENeuroatual/article/view/53/130>>. Acesso em: 2010.

MURPHY-CHUTORIAN, E.; TRIVEDI, M. M. Head Pose Estimation in Computer Vision: A Survey. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, v. 31, n. 4, p. 607-626, 2009.

NACKE, L. E. *et al.* Biofeedback game design: using direct and indirect physiological control to enhance game interaction. *Proceedings of the 2011 annual conference on Human factors in computing systems*. Vancouver, BC, Canada: ACM, 2011. p. 103-112.

NINTENDO. Nintendo Wii Fit. Disponível em: <<http://wiifit.com/>>. Acesso em: 15/02/2012.

OMNIVISION TECHNOLOGIES. OV2640 2 MPixel Product Brief. Disponível em: <http://www.ovt.com/download_document.php?type=sensor&sensorid=3>. Acesso em: 29/07/2011.

OVERMARS, M. Teaching computer science through game design. **Computer**, v. 37, n. 4, p. 81-83, 2004.

PAUSCH, R.; PROFFITT, D.; WILLIAMS, G. Quantifying immersion in virtual reality. *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*: ACM Press/Addison-Wesley Publishing Co., 1997. p. 13-18.

PIERCE, E. HSV Cone. Disponível em: <http://upload.wikimedia.org/wikipedia/commons/f/f1/HSV_cone.jpg>. Acesso em: 29/11/2011.

PIZZI, A. *et al.* Evaluation of upper-limb spasticity after stroke: A clinical and neurophysiologic study. **Arch Phys Med Rehabil**, v. 86, n. 3, p. 410-5, Mar 2005.

ROBERTSON, G.; CZERWINSKI, M.; DANTZICH, M. V. Immersion in desktop virtual reality. *Proceedings of the 10th annual ACM symposium on User interface software and technology*. Banff, Alberta, Canada: ACM, 1997. p. 11-19.

SAXE, D.; FOULDS, R. Toward robust skin identification in video images. In: *Automatic Face and Gesture Recognition*, 1996., Proceedings of the Second International Conference on, 14-16 Oct 1996. 1996. p.379-384.

SCHALLERT, T.; LEASURE, J. L.; KOLB, B. Experience-associated structural events, subependymal cellular proliferative activity, and functional recovery after injury to the central nervous system. **J Cereb Blood Flow Metab**, v. 20, n. 11, p. 1513-28, Nov 2000.

SECOLI, R. *et al.* Effect of visual distraction and auditory feedback on patient effort during robot-assisted movement training after stroke. **J Neuroeng Rehabil**, v. 8, p. 21, 2011.

SEDLÁČEK, M. Evaluation of RGB and HSV models in Human Faces Detection. **Central European Seminar on Computer Graphics**, v. 8, p. 125-131, 2004.

SHIN, M. C.; CHANG, K. I.; TSAP, L. V. Does Colorspace Transformation Make Any Difference on Skin Detection? **Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision**, v., p. 275, 2002.

SHUMWAY-COOK, A.; WOOLLACOTT, M. H. **Motor control: translating research into clinical practice**. Lippincott Williams & Wilkins, 2007.

SILVA, T. S.; SILVEIRA, M. S. Antecipando a avaliação de IHC: verificação de diretrizes a partir de modelos. **Proceedings of the VIII Brazilian Symposium on Human Factors in Computing Systems**, v. 8, p. 248-251, 2008.

SKARBNIK, N.; SAGIV, C.; ZEEVI, Y. Y. Edge Detection and Skeletonization using Quantized Localized phase. In: *Proceedings of 17th European Signal Processing Conference*. 2009. p.1542-1546.

SONY. PlayStation Move. Disponível em: <<http://br.playstation.com/ps3/playstation-move/>>. Acesso em: 24/02/2012.

STONE, M. C. **A Field Guide to Digital Color**. AK Peters, 2003.

SUMA, E. A. *et al.* FFAST: The Flexible Action and Articulated Skeleton Toolkit. In: *Virtual Reality Conference (VR)*, 2011 IEEE, 19-23 March 2011. 2011. p.247-248.

SUTHERLAND, I. E. Sketch pad a man-machine graphical communication system. *Proceedings of the SHARE design automation workshop*: ACM, 1964. p. 6.329-6.346.

_____. A head-mounted three dimensional display. *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. San Francisco, California: ACM, 1968. p. 757-764.

TANENBAUM, A. S. **Organização Estruturada de Computadores**. 5. ed. São Paulo: Prentice Hall, 2006.

TRAVIS, D. **Effective color displays: theory and practice**. Academic Press, 1991.

VEZHNEVETS, V.; SAZONOV, V.; ANDREEVA, A. A Survey on Pixel-Based Skin Color Detection Techniques. In: *Proceedings of the GraphiCon 2003*. 2003. p.85-92. Disponível em:<<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.5.521>>. Acesso em:

WANG, J.-G.; SUNG, E. Frontal-view face detection and facial feature extraction using color and morphological operations. **Pattern Recognition Letters**, v. 20, n. 10, p. 1053-1068, 1999.

WANG, R. Y.; POPOVI, J. Real-time hand-tracking with a color glove. **ACM Trans. Graph.**, v. 28, n. 3, p. 1-8, 2009.

WICKENS, C. D.; BAKER, P. Cognitive issues in virtual reality *Virtual environments and advanced interface design.*: New York, NY, US: Oxford University Press, 1995. p. 514-541.

ZARIT, B. D.; SUPER, B. J.; QUEK, F. K. H. Comparison of five color models in skin pixel classification. In: Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 1999. Proceedings. International Workshop on, 1999. 1999. p.58-63.

ZHOU, H.; HU, H. Human motion tracking for rehabilitation - A survey. **Biomedical Signal Processing and Control**, v. 3, n. 1, p. 1-18, 2008.

ZHOU, W.; BOVIK, A. C. Mean squared error: Love it or leave it? A new look at Signal Fidelity Measures. **Signal Processing Magazine, IEEE**, v. 26, n. 1, p. 98-117, 2009.

ZIMMERMAN, T. G. *et al.* A hand gesture interface device. **SIGCHI Bull.**, v. 18, n. 4, p. 189-192, 1986.

ZIMMERMAN, T. G.; LANNIER, J. Z. *Computer Data Entry and Manipulation Apparatus and Method.* United States of America, 1991.