

Benigno Alonso Gomez

**Sistema de Controle por Imagem para
Indivíduos Tetraplégicos ou com Lesões
Cerebrais**

Dissertação apresentada ao Programa de Pós-Graduação Interunidades em Bioengenharia – Escola de Engenharia de São Carlos, Faculdade de Medicina de Ribeirão Preto e Instituto de Química de São Carlos da Universidade de São Paulo como parte dos requisitos para obtenção do título de Mestre em Bioengenharia.

Área de Concentração: Bioengenharia

Orientador : Prof. Dr. Orivaldo Lopes da Silva

São Carlos
2007

Agradecimentos

Agradeço ao Prof. Dr. Orivaldo Lopes da Silva pela sua orientação, incentivo e amizade no decorrer deste trabalho.

Ao Prof. Dr. José Carlos Pereira pela oportunidade de desenvolver este trabalho na Bioengenharia.

Aos amigos Rodrigo “Farelo” Ortolan, Nélson “neRso” Ferreira Júnior, Fernando César “Sasquatch” Fortuna e Nicola “Carcamano” Bloise pela força nos momentos mais difíceis dessa empreitada.

Aos Professores Valentin Obac Roda, Carlos Dias Maciel e Glauco Augusto de Paula Caurin pelas conversas e esclarecimentos.

A meus pais Manuel e Maria Esther, irmão Mario Cesar e avós Manuel e Esther pelo apoio em toda minha vida.

Ao Prof. Dr. José Manoel Jansen da Silva pela ajuda no início desta jornada. E finalmente, à minha querida esposa Ursula Jansen, pelo apoio incondicional, mesmo quando indo contra a sua opinião, em todas as horas e situações.

Resumo

Gomez, B.A. (2006). *Sistema de Controle por Imagem para Indivíduos Tetraplégicos ou com Lesões Cerebrais*. Dissertação - Programa de Pós-Graduação Interunidades em Bioengenharia (PPGIB) da Universidade de São Paulo, São Carlos, 2006.

Um dos grandes desafios para a reabilitação de indivíduos que sofreram traumas resultando na sua incapacidade de interação com o ambiente é o desenvolvimento de um sistema que possibilite a esse indivíduo a mínima capacidade de comunicação. Em casos extremos, indivíduos perdem os movimentos dos membros superiores, pescoço e a capacidade de comunicação por voz, ficando praticamente isolados do ambiente. O sistema de controle por imagem proposto é baseada na interpretação de seqüências de imagens que compõem movimento do globo ocular, usando como elemento de controle a região da pupila iluminada por diodos emissores de luz (LED's) de infravermelho. A tomada de decisão do sistema para a interpretação do movimento do globo ocular utiliza um controlador baseado em lógica *fuzzy*, que classifica o movimento do globo ocular em padrões pré-determinados e gera um sinal de controle, que pode ser utilizado para o acionamento de um *hardware* ou *software* adicionais, conforme a utilização do sistema proposto. O objetivo do presente trabalho é o desenvolvimento de uma interface homem-máquina de baixo custo, permitindo acesso desses indivíduos ao computador pessoal.

Palavras-chave: visão computacional, controle por imagem, lógica *fuzzy*, interface homem-máquina.

Abstract

Gomez, B.A. (2006). *Sistema de Controle por Imagem para Indivíduos Tetraplégicos ou com Lesões Cerebrais*. Dissertação - Programa de Pós-Graduação Interunidades em Bioengenharia (PPGIB) da Universidade de São Paulo, São Carlos, 2006.

One of the biggest challenges for rehabilitation of people that suffered traumas resulting in environment iterating inabilities is a system development that allows this individuals the minimal communication ability. In extreme circumstances, persons loose superior limbs and neck movements, and also voice communication capacity, staying almost isolated from environment. The proposed image controlled system is based on the interpretation of image sequences that compose the eye movement, using as control element the opening of the pupil illuminated by infra-red light emissor diode (LEDs). The decision-making for the eye movement analysis uses a fuzzy logic controller, that classifies this movement accord to pre-determinated patterns standard and returns a control signal, used for a hardware or software activation. The project goal is the development of a low cost man machine interface, allowing the access for these individuals to a personal computer.

Key words: computer vision, imaging control, fuzzy logic, man machine interface.

Lista de Figuras

FIGURA 1 – Interface Homem-Computador Controlado por Infravermelho	12
FIGURA 2 – Método com <i>joystick</i> controlado pelo movimento da cabeça	13
FIGURA 3 – Transmissor de sinais	14
FIGURA 4 – Barra de ferramentas do software CameraMouse	15
FIGURA 5 – Método Eletro-oculografia	16
FIGURA 6 – Implementação de eletro-oculografia com máscara	16
FIGURA 7 – Paciente do Projeto EagleEyes	17
FIGURA 8 – Digitalização de imagens	19
FIGURA 9 – Câmera analógica utilizada	21
FIGURA 10 – Placa de Aquisição de vídeo PCI 1409	23
FIGURA 11 – Aquisição de imagem do sistema no modo normal	23
FIGURA 12 – Aquisição de imagem do sistema no modo invertido	24
FIGURA 13 – Aquisição de imagem do sistema no modo logarítmico	24
FIGURA 14 – Aquisição de imagem do sistema no modo logarítmico invertido	24
FIGURA 15 – Diagrama de Blocos principal	25
FIGURA 16 – Anatomia dos músculos do globo ocular esquerdo	26
FIGURA 17 – Olho projetado 45 graus	27
FIGURA 18 – Projeções do círculo inscrito na área da pupila	29
FIGURA 19 – Exemplo de algoritmo	31
FIGURA 20 – Algoritmo do <i>software</i> proposto	37
FIGURA 21 – Posições finais das trajetórias diferentes	38
FIGURA 22 – Projeção e divisão em quadrantes	39
FIGURA 23 – Posicionamento da origem do controlador <i>fuzzy</i>	40
FIGURA 24 – Divisão da área de atuação do controlador	42
FIGURA 25 – Variável lingüística X	43
FIGURA 26 – Variável lingüística Y	43
FIGURA 27 – Variáveis <i>fuzzy</i> na área de interesse	44
FIGURA 28 – Divisão da área de interesse por $X > Y$	45
FIGURA 29 – Divisão da área de interesse por $X + Y > N + 1$	45
FIGURA 30 – Divisão de decisão do quadrante superior direito	46
FIGURA 31 – Divisão de decisão do quadrante inferior esquerdo	46

FIGURA 32 – Divisão de decisão do quadrante superior direito	47
FIGURA 33 – Divisão de decisão do quadrante inferior direito	47
FIGURA 34 – Variável lingüística X>Y	48
FIGURA 35 – Tela principal do programa	49
FIGURA 36 – Algoritmo para acionamento dos botões	50
FIGURA 37 – Tela principal com botões desabilitados	51
FIGURA 38 – Linha para analise do nível de cinza sem reflexo	53
FIGURA 39 – Nível de cinza da imagem original	53
FIGURA 40 – Nível de cinza depois do filtro de potência	54
FIGURA 41 – Nível de cinza depois da erosão	54
FIGURA 42 – Nível de cinza depois do filtro gaussiano	55
FIGURA 43 – Linha para analise do nível de cinza sobre reflexo no cristalino	55
FIGURA 44 – Nível de cinza da imagem original com reflexo do cristalino	56
FIGURA 45 – Nível de cinza depois do filtro de potência	56
FIGURA 46 – Nível de cinza depois da erosão	57
FIGURA 47 – Nível de cinza depois do filtro gaussiano	57
FIGURA 48 – Espectro de cinza da imagem	58
FIGURA 49 – Imagem correspondente ao espectro de cinza	58
FIGURA 50 – Nível de cinza selecionados entre 0 e 21	59
FIGURA 51 – Nível de cinza selecionados entre 0 e 47	59
FIGURA 52 – Espectro da imagem original	60
FIGURA 53 – Espectro da imagem após filtro gaussiano	60
FIGURA 54 – Imagem adquirida	62
FIGURA 55 – Imagem após a aplicação da máscara	63
FIGURA 56 – Imagem adquirida pela <i>webcam</i>	64
FIGURA 57 – Plano vermelho da imagem adquirida pela <i>webcam</i>	64
FIGURA 58 – Editor de regras <i>fuzzy</i>	68

Lista de Tabelas

TABELA – 1 – Acertos com o sistema	17
TABELA – 2 – Tempos de processamento com aquisição analógica	65
TABELA – 3 – Tempos de processamento com aquisição analógica e algoritmo otimizado	65
TABELA – 4 – Tempos de processamento com aquisição via <i>webcam</i> (resolução de imagem de 640x480)	66
TABELA – 5 – Tempos de processamento com aquisição via <i>webcam</i> e algoritmo otimizado (resolução de imagem de 640x480)	66
TABELA – 6 – Tempos de processamento com aquisição via <i>webcam</i> (resolução de imagem de 320x240)	67
TABELA – 7 – Tempos de processamento com aquisição via <i>webcam</i> e algoritmo otimizado (resolução de imagem de 320x240)	67
TABELA – 8 – Tempos de Processamento dos diversos algoritmos e sistemas de aquisição	69

Sumário

1 – Introdução	10
2 – Materiais e Métodos	20
2.1 - <i>Hardware</i> de Aquisição	22
2.2 - O movimento do globo ocular	25
2.3 - Localização da Variável de Controle	28
2.4 - Classificação do movimento e determinação da ação	37
2.5 - O Controlador <i>Fuzzy</i>	40
2.6 - Tomada de decisão	48
3 – Resultados	52
3.1 - Implementação dos Filtros	52
3.2 - Tempos de Processamento	64
4 – Discussão	69
5 – Conclusão	71
Bibliografia	73
Apêndice	75

1 - INTRODUÇÃO

A relação do homem com o ambiente onde vive sempre se dá de maneira natural, sem que ele perceba o seu entrosamento com o meio. Desde o nascimento, o homem começa a desenvolver seus sentidos e funções que para todos são normais, tais como a sua movimentação, sua comunicação e sua interatividade com o ambiente e com o próximo. Porém, existem situações em que essas funções, tão naturais para a maioria e que são pouco percebidas no cotidiano, são alijadas do indivíduo. Essa situação, sempre muito traumática para a pessoa, pode ter as mais variadas causas. Neste caso, podemos ter como a raiz do problema lesões neurológicas graves, sendo as suas origens das mais variadas naturezas, como por exemplo, um trauma encéfalo-medular, um acidente vascular, uma degeneração congênita ou não, uma complicação cirúrgica. As perdas de funções (motoras, sensitivas, etc.) podem ser em maior ou menor intensidade, contudo só se pode analisar fisicamente o grau de comprometimento das funções, sendo o impacto psicológico sobre o indivíduo algo muito complexo para ser avaliado. Enfocando as lesões motoras, podemos classificá-las de diversos modos, tais como localizadas, de um dimídio, altas ou baixas, extensas ou não. Em todos os casos, a reabilitação se faz necessária para prover o indivíduo de métodos ou materiais que possam facilitar a adaptação às novas situações que ele encontrará. A reabilitação pode

ser feita por exercícios, pressões e contra-pressões, próteses ou sistemas para adaptação ou substituição de funções perdidas.

No âmbito de graves lesões medulares, causadoras de tetraplegia, e lesões cerebrais, uma das maiores dificuldades do indivíduo é a comunicação, por muitas vezes comprometida tanto pela perda como pela limitação da capacidade de falar. Em alguns casos está preservada a capacidade de falar, porém não há nenhum movimento possível da cabeça ou de outra parte do corpo. Atualmente, em diversas partes do mundo se trabalha para solucionar, de diversas maneiras, essa limitação.

Restringindo os métodos de reabilitação aos sistemas que não usam o controle de voz, temos um campo comum entre lesionados medulares e lesionados cerebrais, independente da causa de suas lesões. Geralmente há grande dificuldade de comunicação, e quando não, muito restrita. A falta de movimento da cabeça, algo que desde pequeno usamos para a simples escolha entre o sim e o não, limita em muito a mais básica forma de comunicação. Nesse caso extremo, temos poucos recursos para possibilitar uma comunicação ou uma interação do indivíduo com o meio.

Em casos menos extremos, onde o indivíduo possui a capacidade de movimentar a cabeça, podemos citar alguns sistemas para comunicação, como por exemplo uma interface homem-computador controlada por infravermelho (CHEN et al,1999). Este sistema baseia-se em um emissor de infravermelho, acoplado ao centro de um par de óculos, um laser de baixa potência também posicionado no centro dos óculos, um conjunto de 3 alvos, onde dois representam duas metades de um teclado convencional e o terceiro alvo é um controle para o posicionamento do mouse, e um sistema de disparo do infravermelho pelo toque da língua, que está montado sobre um fone de ouvido com adaptação para um microfone, onde se localiza o botão de disparo, como podemos ver na figura 1.

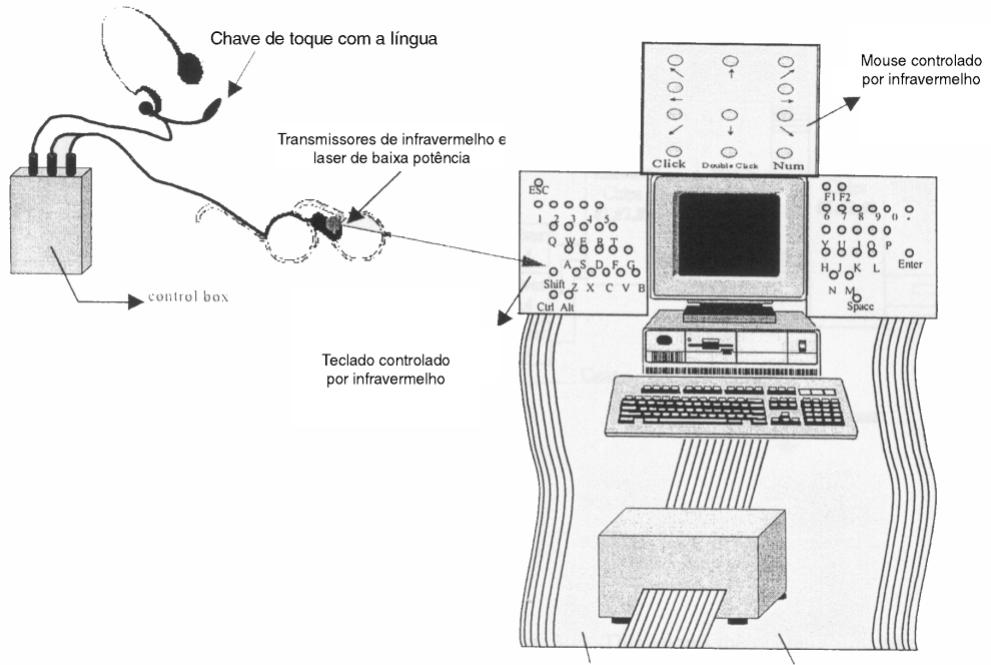


Figura 1 – Interface Homem-Computador Controlado por Infravermelho

(CHEN et al,1999)

Basicamente esse sistema funciona como um sistema de mira a laser. Com o movimento do pescoço, o laser é posicionado sobre o que o indivíduo quer acionar (tecla ou movimento do mouse) e com a língua ele efetua o disparo do infravermelho. Nos alvos existem receptores de infravermelho correspondentes a cada tecla representada e aos oito movimentos possíveis de um mouse (dois verticais, dois horizontais e quatro diagonais). O sistema possui a vantagem de poder simular completamente o uso de um microcomputador, porém o custo é elevado e o acionamento um tanto quanto lento. Além disso, indivíduos que não possuem o movimento do pescoço e seu respectivo controle ficam impossibilitados de utilizar este sistema, visto requerer um certo domínio de movimentos para ser aproveitado ao máximo.

Uma variante desse método, com maior limitação quanto ao número de decisões, mas com maior precisão das mesmas, é um sistema de simulação de um mouse, controlado por joystick operado pelo movimento do pescoço (EVANS;DREW;BLENKHORN,2000). Este sistema utiliza 4 LED's operando na faixa de infravermelho e um sensor de infravermelho, como nas figuras 2 (posição do indivíduo em frente ao receptor) e 3 (indivíduo utilizando o sistema).



Figura 2 – Método com *joystick* controlado pelo movimento da cabeça
(EVANS;DREW;BLENKHORN,2000)



Figura 3 – Transmissor de sinais (EVANS;DREW;BLENKHORN,2000)

O sensor fica posicionado sobre o monitor e o emissor na cabeça do indivíduo. Esse emissor é composto de quatro LED's que ficam posicionados em losango, de forma que a combinação entre o movimento do pescoço, associado a uma seqüência de acionamento dos LED's proporciona a definição do movimento do mouse no monitor. Apesar de ser mais preciso quanto ao movimento do mouse por este ser proporcional à radiação recebida pelo sensor, este sistema, como o anterior, não pode ser utilizado por indivíduos que não tenham o controle do movimento do pescoço. Existe ainda um sistema chamado CameraMouse [internet: <http://www.cameramouse.com>], desenvolvido com tecnologia do Boston College, constituído por um software de controle de mouse sem o uso das mãos, pelo monitoramento de movimentos corporais, podendo ser estes movimentos feitos com a cabeça ou um dedo, convertidos em movimentos do cursor no monitor do computador. O sistema necessário para a operação deste software consiste em um microcomputador e uma câmera com saída USB. Para o

acionamento dos botões do mouse existe uma barra de ferramentas que é mostrada na figura 4.



Figura 4 - Barra de ferramentas do software CameraMouse(Internet -

<http://www.cameramouse.com>)

Note-se que essa interface tem a característica de poder emular os comandos de um mouse. Uma vantagem desse sistema é executar qualquer programa na plataforma Windows, sem haver a necessidade de adaptações no software que será controlado. Por outro lado, não pode ser utilizado por pacientes que possuam sérias restrições de movimentos.

Um outro método, usando eletro-oculografia (LACOURSE; HLUDIK, 1990) (KAUFMAN; BANDOPADHAY; SHAVIV,1993) (YINGXI; NEWMAN, 2004) (WIJESOMA et al,2005) (HORI; SAKANO; SAITO, 2004), é o mais utilizado, por ser de custo baixo e simples utilização. Ele se baseia em monitorar os sinais mioelétricos dos músculos responsáveis pelo movimento do globo ocular. Como podemos ver na figura 5, os eletrodos são dispostos verticalmente sobre um dos olhos e horizontalmente externamente aos olhos. Existe ainda um eletrodo de referência dos potenciais.

Quando os olhos se movimentam, é gerado um potencial em cada par de eletrodos, como podemos ver na figura 5 (LACOURSE;HLUDIK,1990), sendo que a combinação dos sinais disposta em um gráfico cartesiano nos proporciona um ponto correspondente à direção de deslocamento dos globos oculares. Com isso podemos registrar o movimento dos olhos com apenas duas medidas de tensão. Isso reduz em

muito o custo da implementação, mas existe o fato de o indivíduo ter eletrodos posicionados sobre a pele e existirão fios para ligá-los ao aparelho.

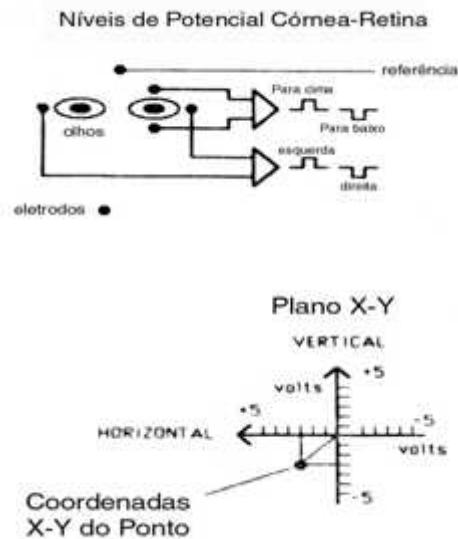


Figura 5 – Método Eletro-oculografia – (LACOURSE;HLUDIK,1990)

Uma evolução nesse conceito é a utilização de uma máscara para melhor fixação dos eletrodos mostrados na figura 6 (KAUFMAN; BANDOPADHAY; SHAVIV,1993). Os resultados são promissores, pois eles possuem um nível alto de acerto de escolhas pelo paciente (tabela 1).



Figura 6 – Implementação de eletro-oculografia com máscara –

<http://www.cs.sunysb.edu/~vislab/projects/eye/poster/index.html>

Tabela 1 - Acertos com o sistema

Ação	% de acertos
Seleção de Menus	73 %
Seleção de Menus (somente 4 posições na tela)	90 %
Detecção Horizontal	75 %
Detecção Horizontal (somente 4 posições na tela)	99 %
Detecção Vertical	92 %
Detecção Vertical (somente 4 posições na tela)	92 %

Uma aplicação já implementada no Boston College é o projeto EagleEyes (Figura 7) (<http://www.bc.edu/schools/csom/eagleeyes>) que utiliza o método da eletro-oculografia em pacientes com diversas patologias neurológicas. A implementação neste caso faz uso de eletrodos para a eletro-oculografia e um sistema de aquisição de dados acoplados a um computador.



Figura 7 – Paciente do Projeto EagleEyes -
[\(http://www.bc.edu/schools/csom/eagleeyes/eagleeyes/gallery/page9/\)](http://www.bc.edu/schools/csom/eagleeyes/eagleeyes/gallery/page9/)

Nos trabalhos aqui expostos existe sempre algum instrumento que depende do deslocamento do indivíduo ou de implantação, como eletrodos, mesmo que não invasivos, ou aparelhos afixados na cabeça. Isso implica, mesmo que em grau muito baixo, a necessidade de monitorização contínua dos instrumentos e medidas de segurança que invariavelmente aumentam o custo de implementação e manutenção.

Este trabalho propõe um sistema de monitoração dos movimentos do globo ocular através de um sistema de aquisição de imagens e tomada de decisão analisando a trajetória de um determinado ponto na imagem.

Entende-se por aquisição de imagem a captura da imagem por um sensor e sua digitalização, ou seja, a conversão dos dados analógicos em um formato que o computador consiga interpretar. Essa imagem é dividida em pixels (*Picture Element*) que possuem valores inteiros e positivos dentro de uma escala determinada pelo número de bits que compõem esse pixel. O número de níveis dessa escala é dado por 2^n onde n é o número de bits. Uma imagem em escala de cinza de 8 bits possui 256 elementos que ao serem representados na tela do computador geram uma escala em tons de cinza variando do branco ao preto com 254 tons diferentes.

Outra característica da digitalização de imagens é a resolução da imagem dada pelo número de pixels que a compõe. Quanto maior o número de pixels maior a definição da imagem digitalizada. Como a imagem é um sinal contínuo e sua digitalização nos fornece uma imagem composta por pixels existe a possibilidade que dentro da área de representação uma grande variação do valor que o pixel deve representar. Nesse caso o pixel assume um valor intermediário. A figura 8 representa a seqüência de digitalização de uma imagem.

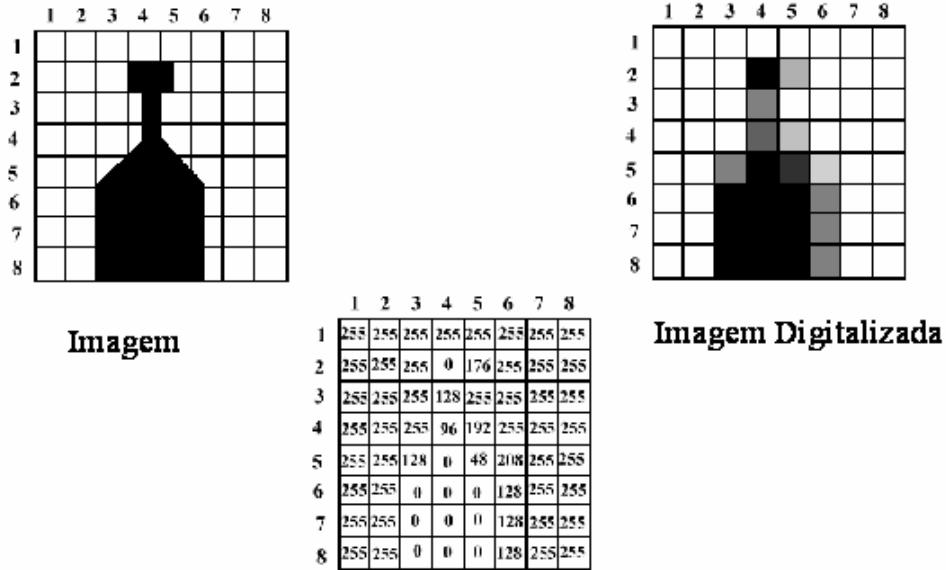


Figura 8 – Digitalização de imagem

No caso de imagens coloridas o modelo RGB (*Red, Green e Blue*) baseia-se no fato de que todas as cores são compostas pelas 3 cores primárias, vermelho, verde e azul. Fazendo analogia ao exemplo da escala de cinza, para cada cor temos 8 bits, sendo o total de 24 bits por pixel. Isso possibilita a representação de 16.777.276 cores diferentes.

A vantagem desse método é ser independente de posições de eletrodos e de movimentação do pescoço, sendo possível a sua aplicação em pacientes com lesões altas da medula, resultante de traumatismo crânio-encefálico (TCE) ou acidente vascular cerebral (AVC).

2 - Materiais e Métodos

A escolha e implementação de um sistema de aquisição de imagens nem sempre é facilmente definida, pois existem inúmeros sistemas, dos mais diversos fabricantes e sobretudo dos mais diversos preços. Algumas características como resolução de aquisição, sistema de aquisição (colorido, monocromático, infravermelho, etc.) e quantidade de *frames* por segundo de aquisição são variáveis que devem ser definidas para chegar a uma relação custo-benefício ótima.

O sistema definido para este trabalho é um sistema de aquisição de imagens utilizando-se uma câmera com LED's de iluminação infravermelha e captura tanto do espectro da luz visível quanto do início do espectro de infravermelho.

Existem no mercado diversos modelos de câmeras que respondem a essa faixa de freqüência, podendo ser analógicas ou digitais. O diferencial básico, além do modo de operação, é o custo de uma câmera que opera nessa faixa. As digitais estão na ordem de US\$ 20,000.00, preço proibitivo para o objetivo do projeto, enquanto câmeras comuns utilizadas em circuitos de segurança, que têm um alcance pequeno porém suficiente, custam em torno de R\$ 80,00. Levando em conta esse fator e a existência de *Digital Signal Processor* (DSP) que trabalham com o padrão NTSC, para que no futuro seja viável a implementação em um sistema embarcado de baixo custo, fez-se a opção de desenvolver um sistema utilizando uma câmera analógica de baixo custo, monocromática e de pequenas dimensões. A câmera escolhida foi um modelo

largamente utilizado em sistemas de circuito fechado de televisão (CFTV) de segurança, como a mostrada na Figura 9.



Figura 9 - Câmera analógica utilizada

Diversas ferramentas de análise na área médica, principalmente na oftalmológica, utilizam câmeras de infravermelho, pois a área da pupila no olho humano não reflete a luz que incide sobre a mesma, produzindo em uma imagem em tons de cinza, uma região cuja componente principal de cinza é o preto e com formato aproximadamente circular. Essa característica é à base do monitoramento do movimento do globo ocular, pois trata-se de uma característica da imagem, no caso uma região aproximadamente circular e de tom escuro, facilmente detectada e mensurada por um algoritmo de reconhecimento de padrões.

A intensidade de luz monocromática de infravermelho emitida pelos LED's do sistema estão abaixo de $0,1 \text{ W/cm}^2$, o que permite uma exposição de até $3 \times 10^4 \text{ s}$ (BAXTER).

Na implementação do software utilizou-se o LabVIEW® da National Instruments, por ser uma ferramenta de programação gráfica de fácil utilização, tanto na interface como na estrutura lógica do *software*; além disso, possui inúmeros recursos de integração com os mais diversos *hardwares* existentes no mercado, módulos de

aquisição de imagem, processamento de imagens e controle (que inclui lógica *fuzzy*). Atualmente, a portabilidade de um *software* desenvolvido em LabVIEW® foi ampliada, pois além de operar em plataformas PC utilizando-se sistemas operacionais Windows e Linux, é possível sua implementação em plataformas baseadas em palmtops, *Field Programming Gate Array* (FPGA) e *Digital Signal Processor* (DSP) da Texas Instruments. Essa inovação possibilita o desenvolvimento em um computador de mesa e sua implementação posterior em um sistema autônomo ou embarcado, ou seja, um sistema que utiliza baterias e não precisa estar conectado a nenhum computador sendo assim mais portátil.

2.1 - *Hardware* de Aquisição

Além da câmera, outro item importante do sistema de aquisição de imagens baseado em um computador é a placa de aquisição de imagens. Geralmente, esse dispositivo tem custo elevado. Após pesquisar as opções oferecidas pelo mercado e levando-se em conta os fatores como custo, tipo de imagem a ser adquirida (analógica e monocromática) e número de canais de entrada, escolheu-se o modelo PCI 1409 da National Instruments, mostrada na figura 10. Este modelo é uma placa de aquisição de imagens analógica, com 4 canais de vídeo monocromáticos ou que utilizando uma câmera com saída RGB adquire imagens coloridas. Podemos citar algumas características, tais como: janela de aquisição variável, tanto na dimensão em pixels com na eliminação de bordas da imagem adquirida, filtros digitais para os padrões de vídeo NTSC ou PAL, escala variável, resolução da imagem de 8 ou 10 bits, diversos modos de *trigger*, filtragem inicial podendo ser inversora, logarítmica, anti-logarítmica,

binária ou binária invertida, taxa de aquisição de 30 quadros por segundo e a possibilidade de diversas dessas propriedades serem configuradas independentemente em cada canal de vídeo.

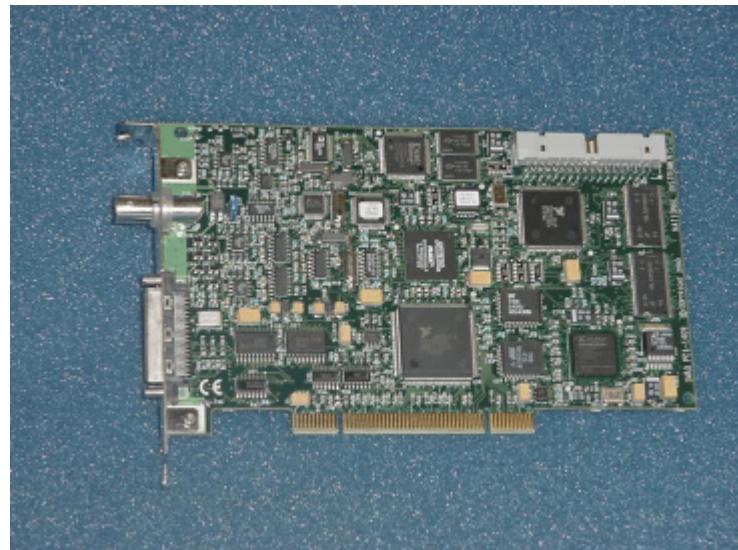


Figura 10 - Placa de Aquisição de vídeo PCI 1409

Outro fator importante para essa escolha é o fato de ser da mesma empresa que produz o *software* LabVIEW®, eliminando completamente o fator de incompatibilidade entre *hardware* e *software* no desenvolvimento do sistema. Nas figuras seguintes são apresentados os resultados relativos de alguns dos modos de aquisição da placa PCI 1409.



Figura 11 - Aquisição de imagem do sistema no modo normal



Figura 12 - Aquisição de imagem do sistema no modo invertido



Figura 13 - Aquisição de imagem do sistema no modo logarítmico



Figura 14 - Aquisição de imagem do sistema no modo logarítmico invertido

Depois de definido o *hardware* de aquisição, o sistema ficou configurado pelo diagrama de blocos da Figura 15, onde os blocos de reconhecimento do movimento/acionamento e análise de movimento foram implementados por *software*.

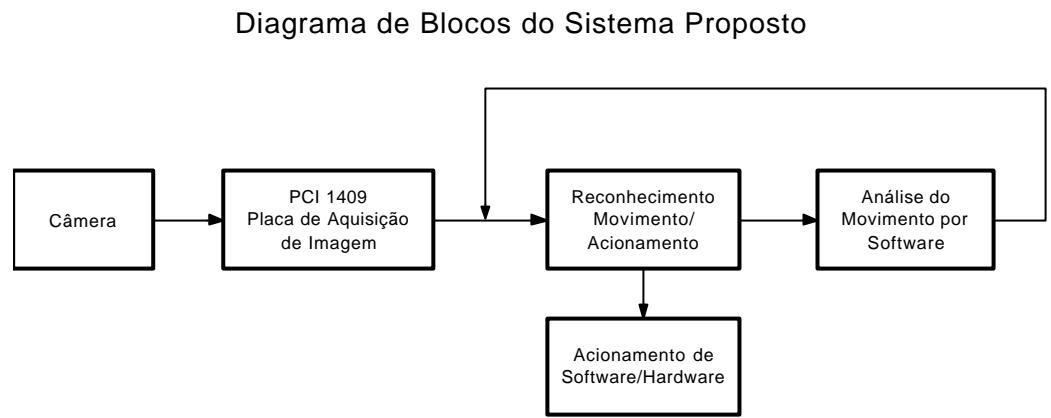


Figura 15 - Diagrama de Blocos principal

2.2 - O movimento do globo ocular

O globo ocular tem seu movimento delimitado pela musculatura motora. Essa musculatura é composta por quatro músculos que permitem o controle de posicionamento e deslocamento do globo ocular e estão dispostos como na Figura 16.

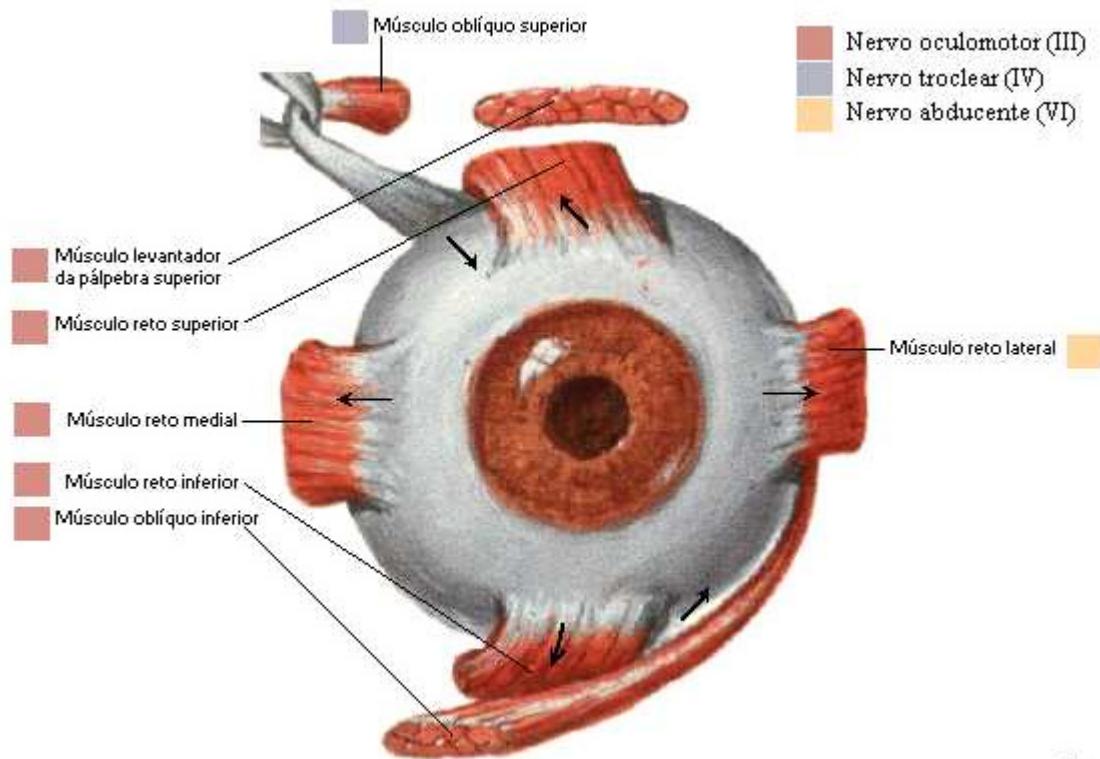


Figura 16 - Anatomia dos músculos do globo ocular esquerdo (adaptado do *software Atlas Interativo da Anatomia Humana*, 1999)

Apesar de possuir movimento limitado, o globo ocular desloca-se horizontalmente e verticalmente, além de realizar pequenas rotações. Algumas patologias causam modificações no padrão de movimento bem como na capacidade de fixar o globo ocular em determinada posição.

Tomando como referência a linha de visada do indivíduo olhando para o horizonte, o conjunto de deslocamento máximo médio de cada olho é X graus externo, Y graus interno, Z graus para cima e W graus para baixo.

Estes limites de deslocamento podem ser considerados como valores máximos de parâmetros em qualquer sistema que monitore o deslocamento do globo ocular ou que localize o ponto no qual está focada a visão.

Existe ainda um movimento de deformação do globo ocular que permite ao indivíduo ajustar a distância focal da visão. Esta característica foi desconsiderada neste trabalho, pois a distância focal no sistema tende a permanecer constante, não influenciando no posicionamento do globo ocular.

Como no sistema projetado a variável de interesse é a região da pupila, a determinação da sua deformação em função do deslocamento do globo ocular na sua projeção no plano de captura da imagem é fundamental. Observando a Figura 17 vemos as projeções do globo ocular para diversas posições. Pode-se notar que apesar de distorcida, a imagem da pupila, considerando um valor médio de 6 mm para o seu diâmetro e o raio médio do globo ocular de 24 mm, é facilmente observável, mostrando que com um sistema com câmera fixa em relação ao globo ocular, é possível a observação da pupila em uma região considerável, no caso, um ângulo sólido de 45 graus.

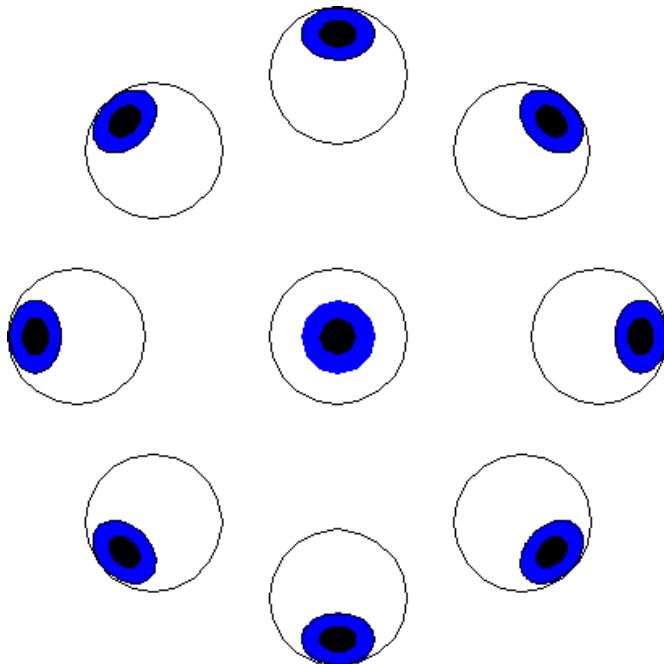


Figura 17 - Olho projetado 45 graus

Conclui-se desse fato que a melhor posição para a câmera é em frente ao globo ocular do indivíduo. Isso não impede que essa posição espacial da câmera seja outra, porém a projeção da pupila no plano da imagem adquirida terá uma distorção não simétrica em relação à posição do globo ocular quando o indivíduo estiver olhando o horizonte. Outra vantagem dessa posição é cobrir a maior área de deslocamento da pupila dentro da área delimitada pelas pálpebras.

2.3 - Localização da Variável de Controle

A partir de uma imagem adquirida foi determinada a posição da pupila nesta imagem. Visto que a região da imagem onde se encontra a pupila é a variável de controle do sistema e que dependendo da posição relativa entre o globo ocular e a câmera esta região possui uma forma circular ou elíptica, optou-se por usar como variável de controle o Centróide da área da pupila ao invés de usar a área total da pupila. No caso desta região possuir uma forma circular, este ponto se localiza no centro da região e no caso da forma elíptica esse ponto se encontra no encontro dos dois eixos. Para a determinação do Centróide na elipse é localizado na região da pupila o maior círculo inscrito dentro da elipse. Assim, independente da posição do globo ocular e consequente forma de projeção da área da pupila no plano da imagem, não há a necessidade de uma modificação no algoritmo da aquisição quanto à estratégia de localizar a área de interesse e seu Centróide, conforme ilustra a figura 18.

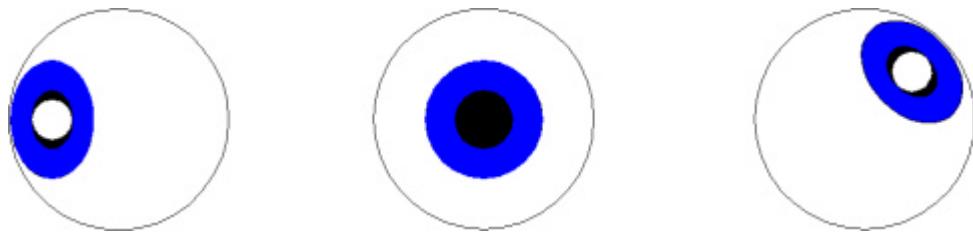


Figura 18 - Projeções do círculo inscrito na área da pupila

Para a localização da área de interesse na imagem, foi necessário um tratamento da imagem adquirida (GONZALEZ;WOODS,1993). Para o tratamento dessa imagem algumas características são necessárias, tais como:

- Ajuste dos valores de brilho e contraste, visto que a área de interesse possui níveis de cinza com valores próximos à zero (preto)
- Limiar para selecionar o intervalo de pixels em escala de cinza
- Análise morfológica da imagem, eliminando objetos nas bordas e pequenas interferências
- Localização do Centróide da pupila

Para esse fim foi elaborado o seguinte algoritmo para o tratamento dessa imagem, possui as seguintes etapas:

- Filtro de potência
- Erosão da imagem
- Filtro gaussiano
- Filtro de nível de cinza
- Eliminação de bordas da imagem
- Eliminação de pequenos objetos da imagem de bordas da imagem
- Preenchimento dos buracos
- Localização de áreas circulares
- Cálculo do Centróide

As funções dessas etapas são descritas a seguir:

- Filtro de potência – Decrementa o brilho e incrementa o contraste em regiões mais claras da imagem e diminui o contraste nas regiões mais escuras da imagem;
- Erosão da imagem – Diminui o tamanho de um objeto ao longo de sua borda e elimina pontos isolados da imagem.
- Filtro gaussiano – Atenua as variações da intensidade da luz na vizinhança de um pixel usando um kernel gaussiano (apêndice 1) na operação de filtragem.
- Filtro de nível de cinza – Com esse filtro seleciona-se a área de interesse variando-se o nível de cinza desse filtro. Trata-se de um filtro passa-faixa cujos valores são definidos pelo usuário em função da luminosidade do ambiente, e dentro desta faixa estão os níveis de cinza da região de interesse. A sua resposta é uma imagem binarizada, onde as regiões escolhidas ficam selecionadas com a cor vermelha e as demais em preto.
- Eliminação de bordas da imagem – Elimina os elementos de imagem que possuem o mesmo nível de cinza da área de interesse que se localizam nas bordas da imagem.
- Eliminação de pequenos objetos da imagem – Elimina regiões de áreas pequenas determinadas pelo parâmetro do filtro e que possuam o mesmo nível de cinza da área de interesse.
- Preenchimento de buracos – Preenche os buracos dentro das áreas de interesse. Basicamente muda o valor dos pixels internos para o valor da borda do buraco. Só é aplicado quando a borda for uma área determinada pelo filtro de nível de cinza.
- Localização de áreas circulares – Procura nas regiões selecionadas áreas onde se é possível inscrever-se círculos com diâmetros dentro de um intervalo pré-

determinado. Os limites inferior e superior dos diâmetros são determinados pelo usuário de tal forma que englobe as projeções mínima e máxima da pupila, sendo estas dependentes da distância do globo ocular ao plano da imagem e do tamanho do globo ocular.

- Cálculo do Centróide – Calcula o Centróide das regiões de interesse selecionadas.

Um exemplo desse algoritmo é mostrado na Figura 19.



Figura 19-a - Imagem adquirida

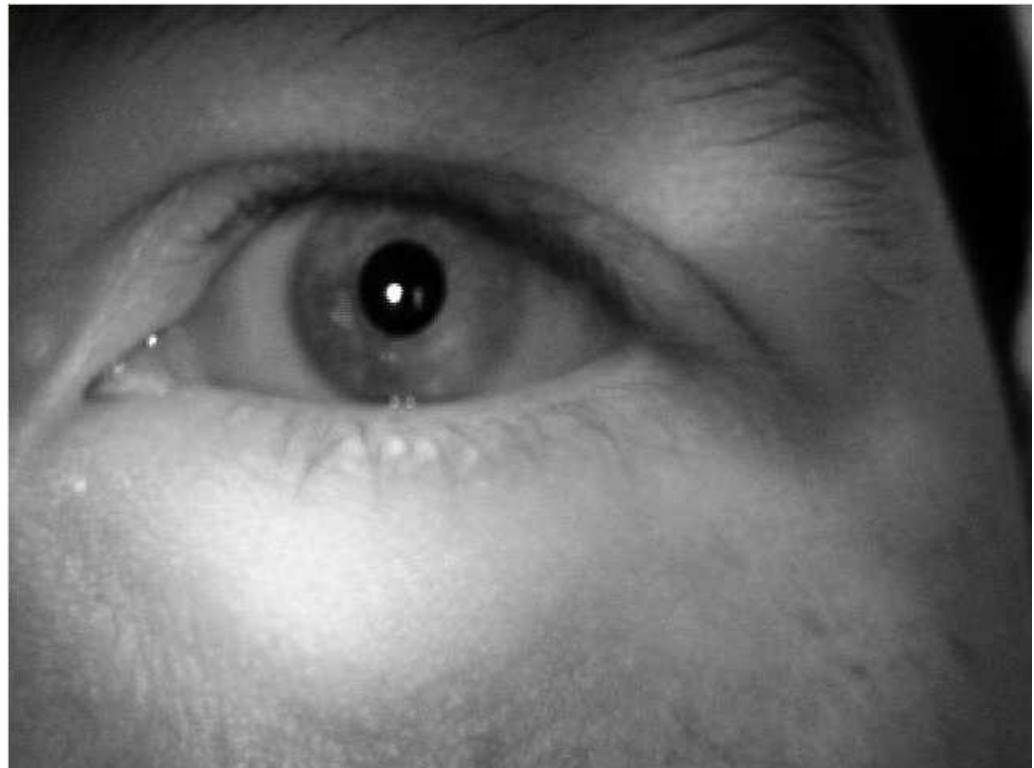


Figura 19-b - Filtro de potência



Figura 19-c - Filtro de erosão



Figura 19-d - Filtro gaussiano



Filtro 19-e - Filtro de nível de cinza



Figura 19-f - Eliminação das bordas da imagem

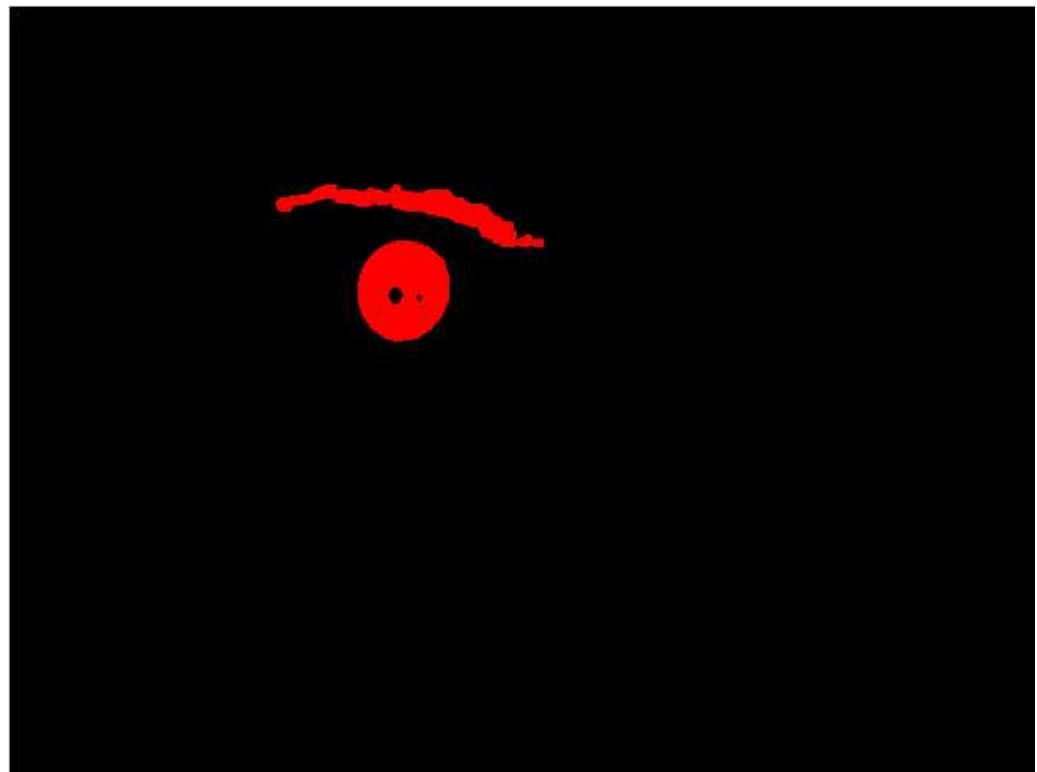


Figura 19-g - Eliminação de pequenos objetos da imagem



Figura 19-h - Preenchimento dos buracos

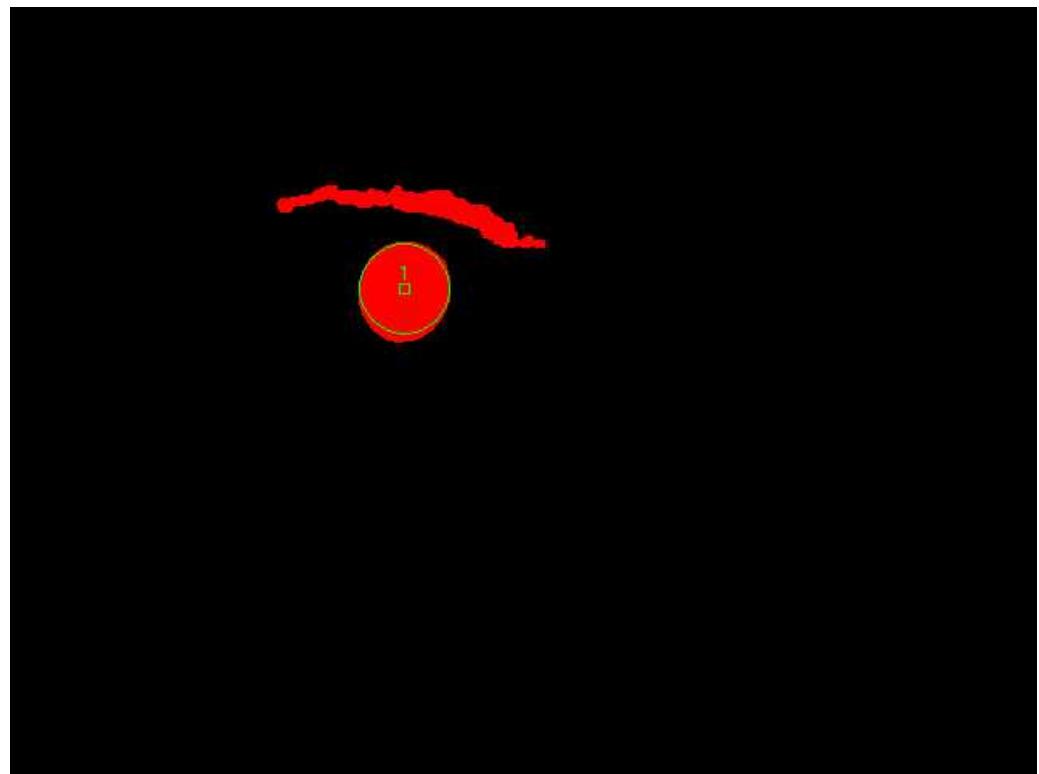


Figura 19-i - Localização das áreas circulares

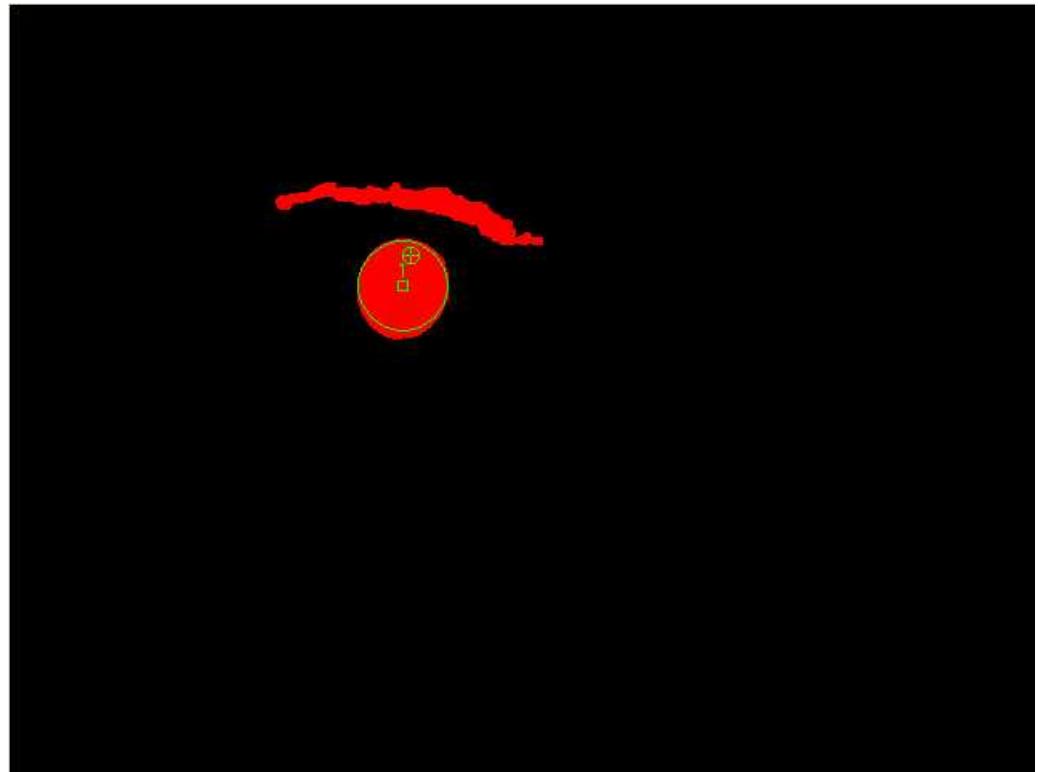


Figura 19-j - Cálculo do Centróide

Este procedimento resulta nas coordenadas de um ponto que localiza o Centróide na imagem adquirida. O algoritmo para o tratamento da imagem é mostrado na Figura 20.

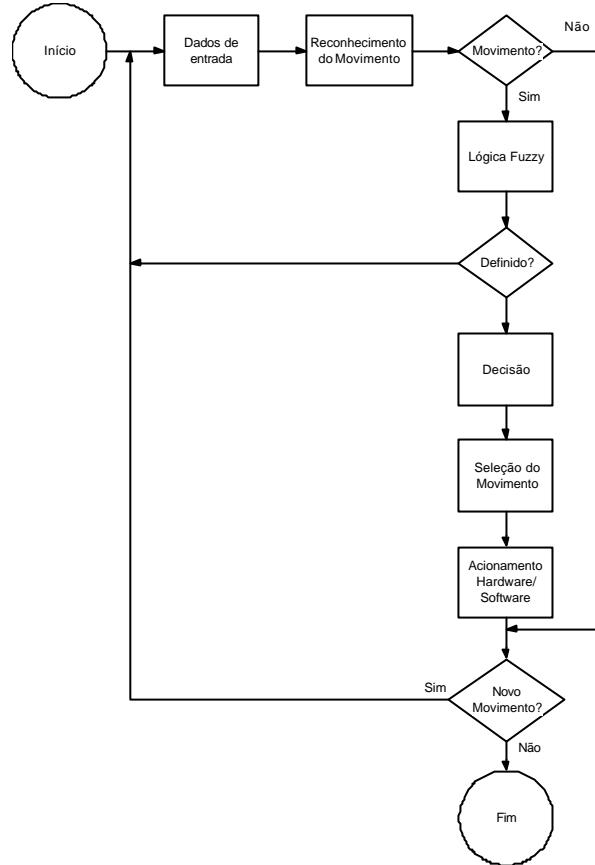


Figura 20 - Algoritmo do *software* proposto

2.4 - Classificação do movimento e determinação da ação

O movimento do globo ocular e consequentemente da pupila, apesar de estarem limitados a uma região determinada pela musculatura, possui liberdade de execução dentro dessa região. Para interpretar todos os movimentos possíveis e classificá-los a fim de selecionar a ação no computador é uma tarefa que demanda muito custo computacional e algoritmos complexos, tornando inviável a sua realização a baixo custo.

Por esse motivo, decidiu-se reduzir a quatro o número de movimentos a serem monitorados, que são os verticais (2) e os horizontais (2) e um evento, que é o piscar do olho.

Os movimentos podem ser divididos em esquerda e direita no caso horizontal e para cima e para baixo no caso do vertical, possibilitando um maior controle pelo indivíduo. Com esses movimentos é possível a seleção de diversos botões nas telas do computador.

Esse conjunto de movimentos nem sempre são executados com precisão pelo indivíduo que está sendo monitorado. Em um breve estudo realizado, apurou-se que o mesmo indivíduo ao executar o mesmo movimento padronizado do globo ocular, partindo de uma posição central da pupila e deslocando o globo ocular horizontalmente ou verticalmente sem estímulo visual, realizava trajetórias semelhantes porém com as posições finais das trajetórias diferentes, ou seja, se encontram na mesma região porém as suas coordenadas são distintas, como mostra a figura 21.



Figura 21 - Posições finais das trajetórias diferentes.

Uma extração dessa característica é quando os movimentos combinam dois deslocamentos, um horizontal e um vertical, podendo, por exemplo, um movimento para a esquerda ter uma posição final deslocada verticalmente para cima e um

movimento para cima ter a sua posição final deslocada horizontalmente para a esquerda. Em uma estratégia de decisão baseada na diferença entre a posição inicial e a posição final, esses movimentos podem gerar imprecisões na tomada de decisão.

Uma simplificação do conjunto de movimentos, apesar de diminuir as possibilidades de classificação do movimento, possibilita uma implementação mais simples e robusta da estratégia de decisão. A Figura 22 demonstra essa simplificação, onde a área de deslocamento da variável de controle é dividida em nove setores. O setor central é a posição de repouso ou origem e seus adjacentes horizontais e verticais centrais, posições de determinação exata do movimento. O círculo tracejado é o limite de deslocamento da variável de controle, que no caso é o limite da área determinada pelo ângulo sólido de deslocamento máximo da linha de visada no plano da imagem adquirida.

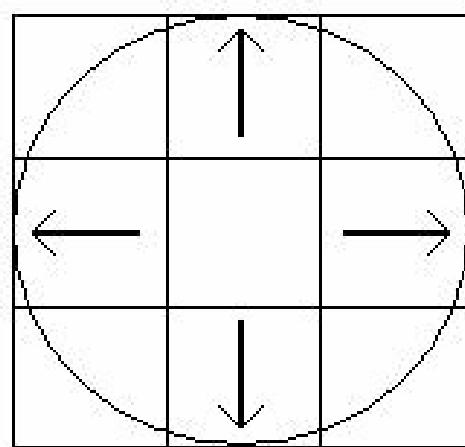


Figura 22 - Projeção e divisão em quadrantes

Esse ângulo é variável para cada indivíduo e deve ser estabelecido de tal maneira que haja um deslocamento grande o suficiente para determinar com precisão o movimento e minimizado ao máximo para que se evite a fadiga da musculatura e esforço desnecessário para realizar o movimento. Como esse ângulo de rotação não é

uniforme para as diversas direções de deslocamento do globo ocular do mesmo indivíduo, são adquiridas imagens em quatro direções (para cima, para baixo, para esquerda e para direita) e calcula-se o deslocamento angular mínimo dessas imagens em relação à posição de repouso, cujo deslocamento com menor ângulo é adotado. A título de exemplo, adotou-se o ângulo de 45°.

Nos quadrantes restantes, ou seja, nos cantos da figura, há a possibilidade de interpretação errônea do deslocamento.

No resto da região dentro dessas áreas o desafio é determinar em qual região de decisão está localizada a variável de controle.

2.5 - O Controlador *Fuzzy*

Ao pesquisar o movimento do globo ocular chega-se à conclusão que cada indivíduo possui o movimento diferente de outro indivíduo. Em alguns casos esses movimentos são parecidos porém em outros casos, o mesmo movimento para dois indivíduos tem trajetórias bem diferentes. Para implementar um sistema de controle robusto com pouca sensibilidade à variação da trajetória adotou-se um sistema de lógica *fuzzy*.

A lógica *fuzzy* (YAGER;FILEV,1994), (VAS,1999), (TSOUKALAS; UHRIG,1997), (SHAW; SIMÕES,1999), (ARZI;MAGNIN,1989) baseia-se em regras do tipo SE < condição1, condição2, ...> ENTÃO <consequência1, consequência2, ...> e em variáveis lingüísticas como um conjunto definido por {FRIO, MORNO, QUENTE} ao invés da temperatura. Essa definição das variáveis lingüísticas necessita da experiência do programador para defini-las de acordo com a necessidade de seu sistema

de controle. Neste caso, a experiência deve levar em conta o movimento natural do indivíduo para a escolha da ação. Na implementação do controlador *fuzzy* adotou-se como variável de entrada as coordenadas do Centróide da figura analisada.

A área de atuação do controlador é determinada a partir da posição de repouso para o globo ocular do indivíduo, ou seja, o indivíduo olhando para o infinito com o plano da imagem paralelo ao eixo vertical da cabeça e perpendicular ao plano medial da cabeça. A fim de se obter a imagem, posiciona-se a câmera à frente do olho que controlará o sistema.

Neste sistema, todas as posições são relativas à posição inicial já obtida. Conseqüentemente após a captura da primeira imagem e seu processamento, a posição (X, Y) do Centróide dessa imagem é considerada a posição central do controlador *fuzzy*. Essa posição pode ser transladada para posição (0,0) onde as variáveis X e Y normalizadas pelo tamanho da figura (N) teriam valores positivos e negativos ou, normalizadas pelas distâncias máximas e transladadas ao centro do controlador, elas teriam o valor (0,5;0,5). Para esta implementação, optou-se pela translação ao centro da área do controlador. Na figura 23 podemos ver essas duas opções.

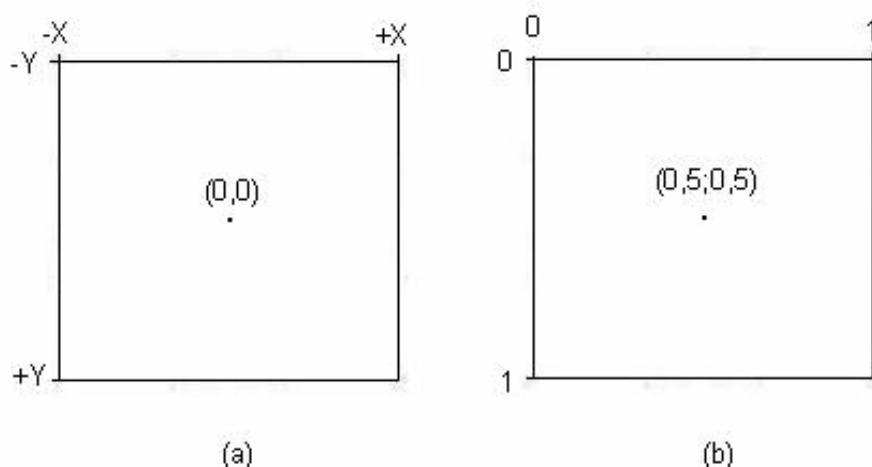


Figura 23 - Posicionamento da origem do controlador fuzzy.

- (a) – no centro da imagem
- (b) – no canto superior esquerdo da imagem

Na implementação do controlador *fuzzy* a área foi dividida em cinco regiões: Esquerda, Direita, Centro, Alto e Baixo. Essa divisão é vista na figura 24.

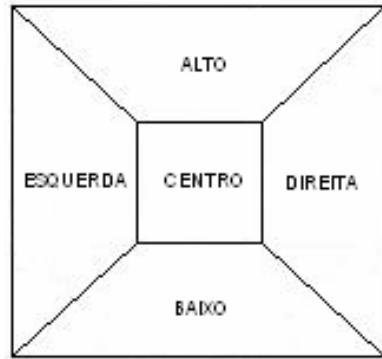


Figura 24 - Divisão da área de atuação do controlador

Sobrepondo à área de atuação uma divisão em quadrantes podemos notar que dos nove quadrantes, cinco não possuem conflitos na tomada de decisão. Porém nos cantos existem duas decisões possíveis que podem ser tomadas. Considerando a origem da imagem o canto superior esquerdo, definimos as variáveis X e Y, correspondentes às coordenadas da imagem, da seguinte forma : X aumenta para a esquerda e Y aumenta para baixo. Essa definição garante a coerência entre o controlador e o sistema de aquisição de imagem. Sendo assim, foram definidas assim as variáveis:

X – conjunto {ESQUERDA, CENTRO, DIREITA}

Y – conjunto {ALTO, CENTRO, BAIXO}

Podemos ver essa implementação nas figuras 25 e 26.

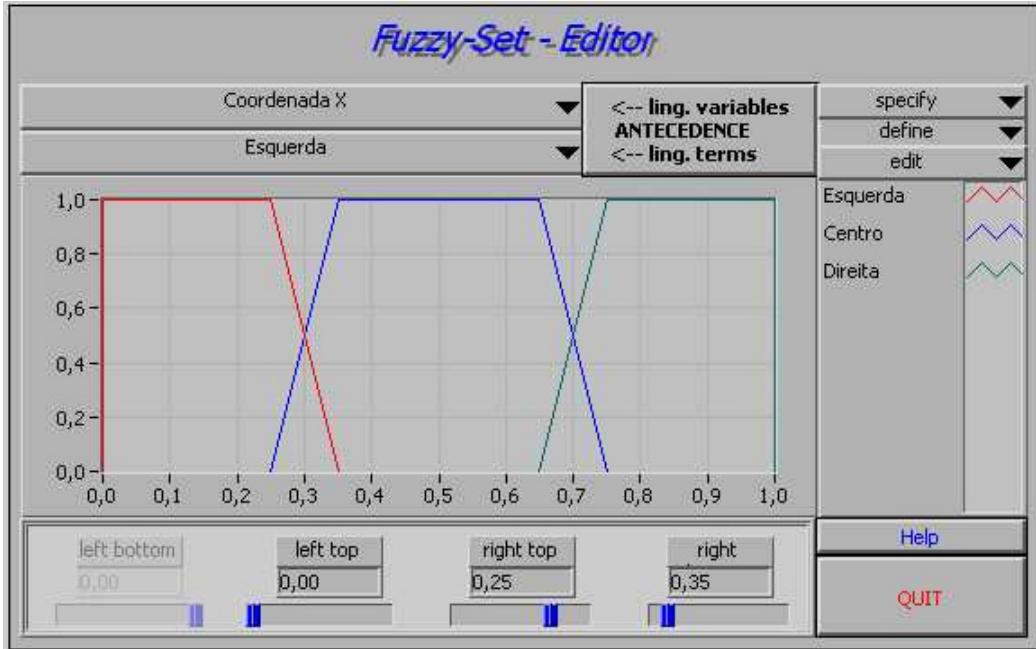


Figura 25 - Variável lingüística X – Tela do *Fuzzy Logic Controller Design*, aplicativo pertencente ao software LabVIEW

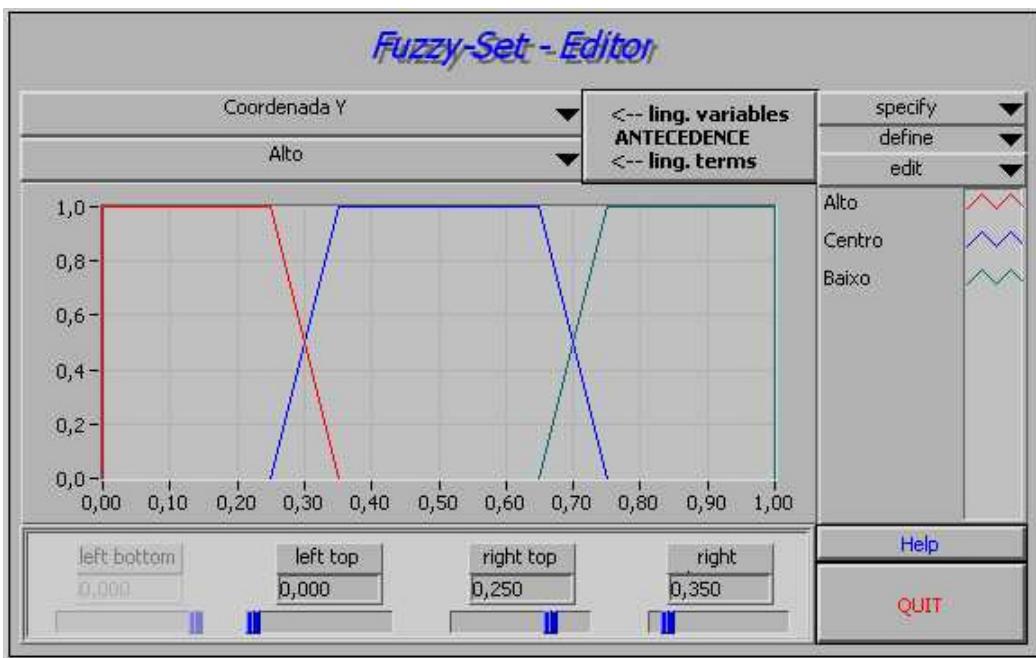


Figura 26 - Variável lingüística Y – Tela do *Fuzzy Logic Controller Design*, aplicativo pertencente ao software LabVIEW

Na área de interesse, a implementação fica como mostrado na figura 27.

X=ESQ Y=ALTO	X=CENT Y=ALTO	X=DIR Y=ALTO
X=ESQ Y=CENT	X=CENT Y=CENT	X=DIR Y=CENT
X=ESQ Y=BAIXO	X=CENT Y=BAIXO	X=DIR Y=BAIXO

DIR = DIREITA
 ESQ = ESQUERDA
 CENT = CENTRO
 ALTO = ALTO
 BAIXO = BAIXO

Figura 27 - Variáveis fuzzy na área de interesse.

Esse tipo de controlador implementado como a área de interesse sendo um quadrado possui a característica de ser adaptável tanto na forma (podendo ser um retângulo qualquer) como na sua divisão interna, mudando as diagonais e o tamanho das áreas de decisão. Isso permite um grande número de combinações permitindo a utilização do sistema até por indivíduos que possuam alguma restrição no movimento do globo ocular, impossibilitando sua total movimentação, utilizar-se deste sistema.

Apesar de sua simplicidade (a implementação com duas variáveis, as coordenadas X e Y), o controlador não consegue identificar nos quadrantes dos cantos qual foi a intenção do indivíduo. Para solucionar esse problema, foram implementadas no controlador *fuzzy* mais duas variáveis. A primeira chamada X>Y divide a área de interesse em duas regiões delimitadas pela diagonal principal da matriz de pontos da imagem. A área de interesse fica delimitada como na figura 28.

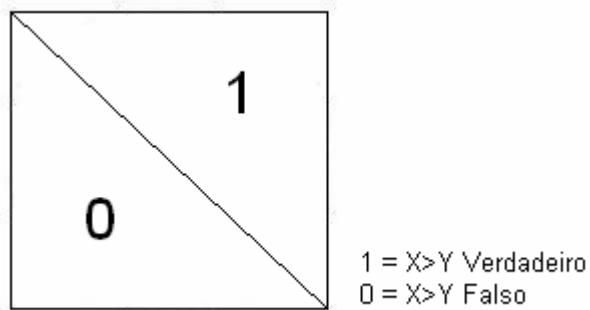


Figura 28 - Divisão da área de interesse por X>Y.

A outra variável foi definida como $X+Y>N+1$, onde N é o tamanho da área de interesse em pixels (no caso, uma imagem de dimensão $N \times N$ pixels). Nos sistemas de aquisição de imagens baseados em CCD as dimensões da imagem adquirida são diferentes ($M \times N$ pixels) mas neste sistema essas dimensões são configuráveis permitindo imagens quadradas ($N \times N$ pixels). Essa variável divide a região de interesse em duas utilizando a diagonal secundária como limite conforme a figura 29.

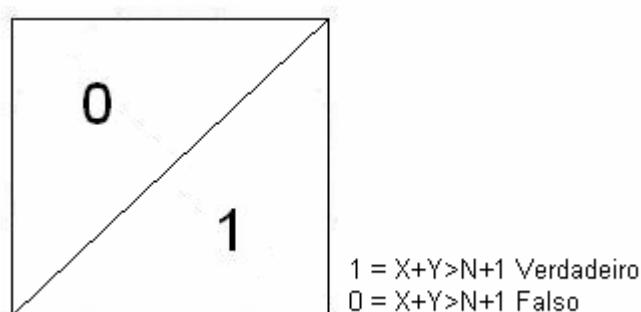


Figura 29 - Divisão da área de interesse por $X+Y>N+1$.

Com essa implementação determinamos as tomadas de decisões nos quadrantes dos cantos. No quadrante superior esquerdo temos os valores de $X =$ Esquerda ; $Y =$ Alto ; $X>Y$ dividindo o quadrante em duas áreas. Se $X>Y$ for verdadeira, temos como consequência a saída para cima. Caso contrário temos como saída a decisão para a esquerda. A figura 30 demonstra como fica a divisão do quadrante.

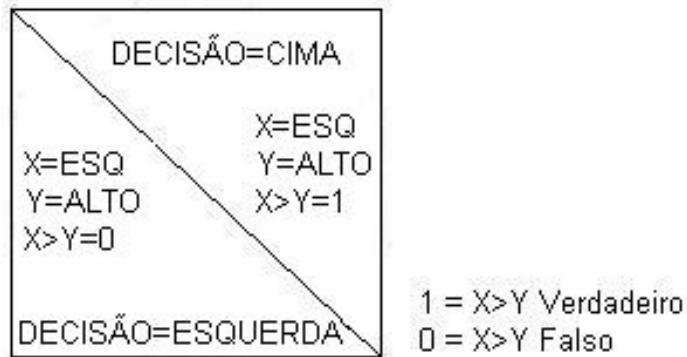


Figura 30 - Divisão de decisão do quadrante superior esquerdo

No quadrante inferior esquerdo temos os valores de $X = \text{Esquerda}$; $Y = \text{Baixo}$; $X+Y>N+1$ dividindo o quadrante em duas áreas. Se $X+Y>N+1$ for verdadeira, temos como consequência a saída para baixo e caso contrário temos como saída a decisão esquerda. A figura 31 demonstra essa divisão do quadrante.

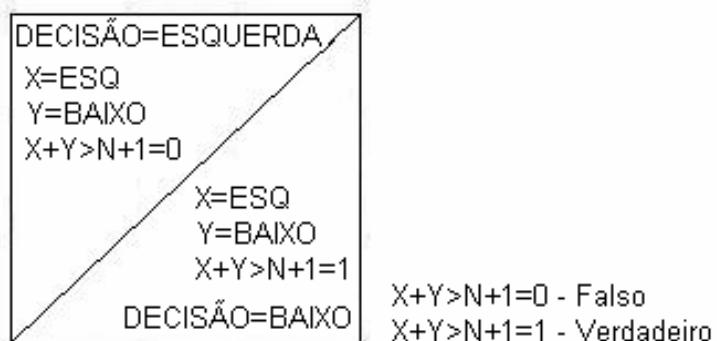


Figura 31 - Divisão de decisão do quadrante inferior esquerdo.

No quadrante superior direito temos os valores de $X = \text{Direita}$; $Y = \text{Alto}$; $X+Y>N+1$ dividindo o quadrante em duas áreas. Se $X+Y>N+1$ for verdadeira, temos como consequência a saída para direita e caso contrário temos como saída a decisão cima. A figura 32 demonstra essa divisão do quadrante.

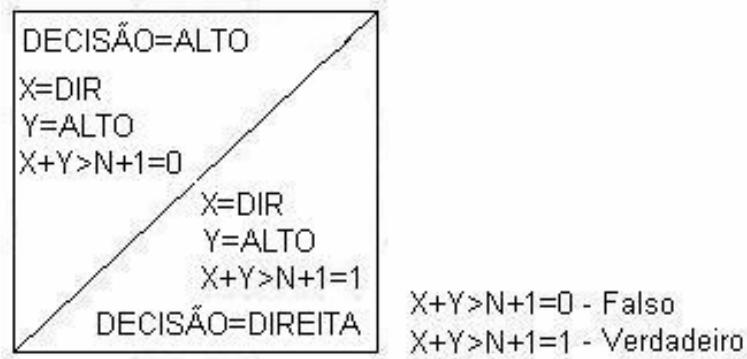


Figura 32 - Divisão de decisão do quadrante superior direito.

No quadrante inferior direito temos os valores de $X = \text{Direita}$; $Y = \text{Baixo}$; $X > Y$ dividindo o quadrante em duas áreas. Se $X > Y$ for verdadeira, temos como consequência a saída para direita e caso contrário temos como saída a decisão baixo. A figura 33 demonstra essa divisão do quadrante.

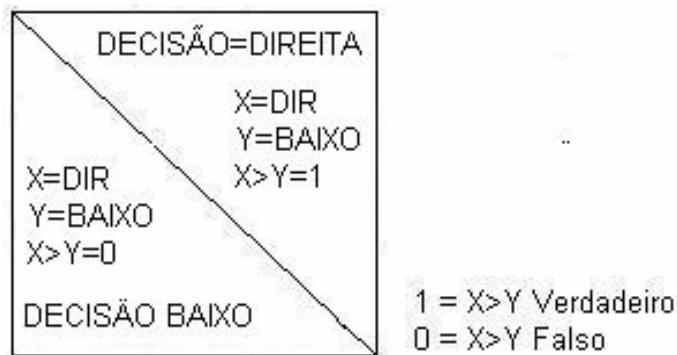


Figura 33 - Divisão de decisão do quadrante inferior direito.

Um exemplo de implementação das variáveis lingüísticas, no caso $X > Y$, é mostrada na figura 34.

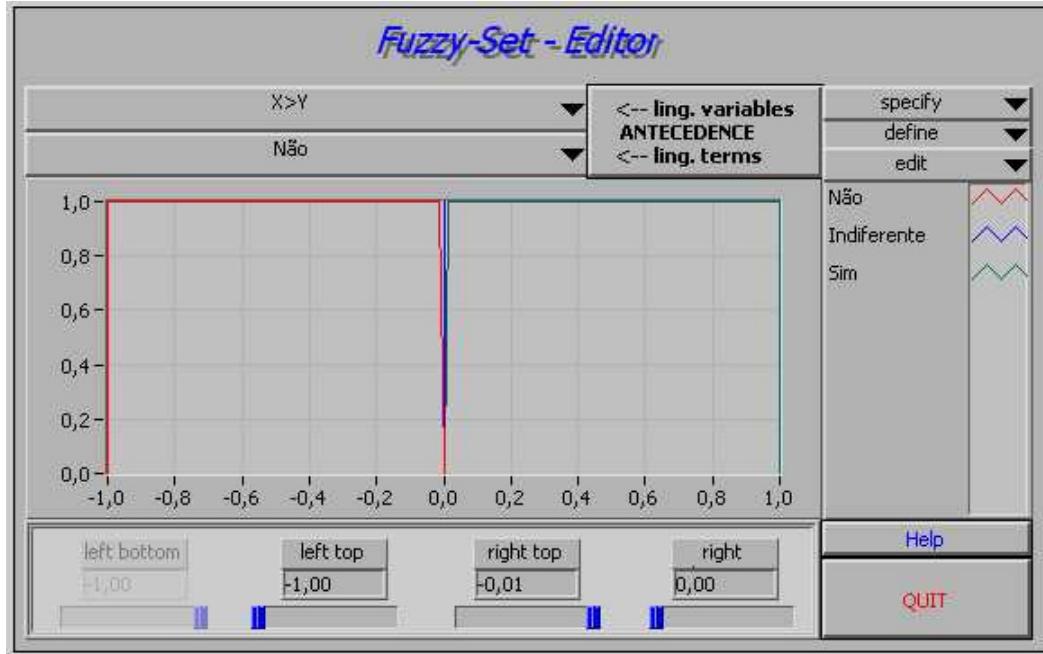


Figura 34 - Variável lingüística $X > Y$ – Tela do *Fuzzy Logic Controller Design*, aplicativo pertencente ao *software LabVIEW*

2.6 - Tomada de decisão

Até este ponto, todas as análises foram feitas em imagens isoladas. Porém, o sistema deve analisar uma seqüência de imagens para tomar uma decisão. Essa decisão pode ser um movimento de deslocamento para escolha de um botão ou um evento, o acionamento do botão selecionado. Para se ter um parâmetro para testes foi implementada uma interface no computador com nove botões, dispostos em três colunas de três botões cada. Esses botões não foram associados a nenhum *software* ou *hardware* na fase de desenvolvimento mas sua associação é de fácil implementação. Essa interface é mostrada na figura 35.

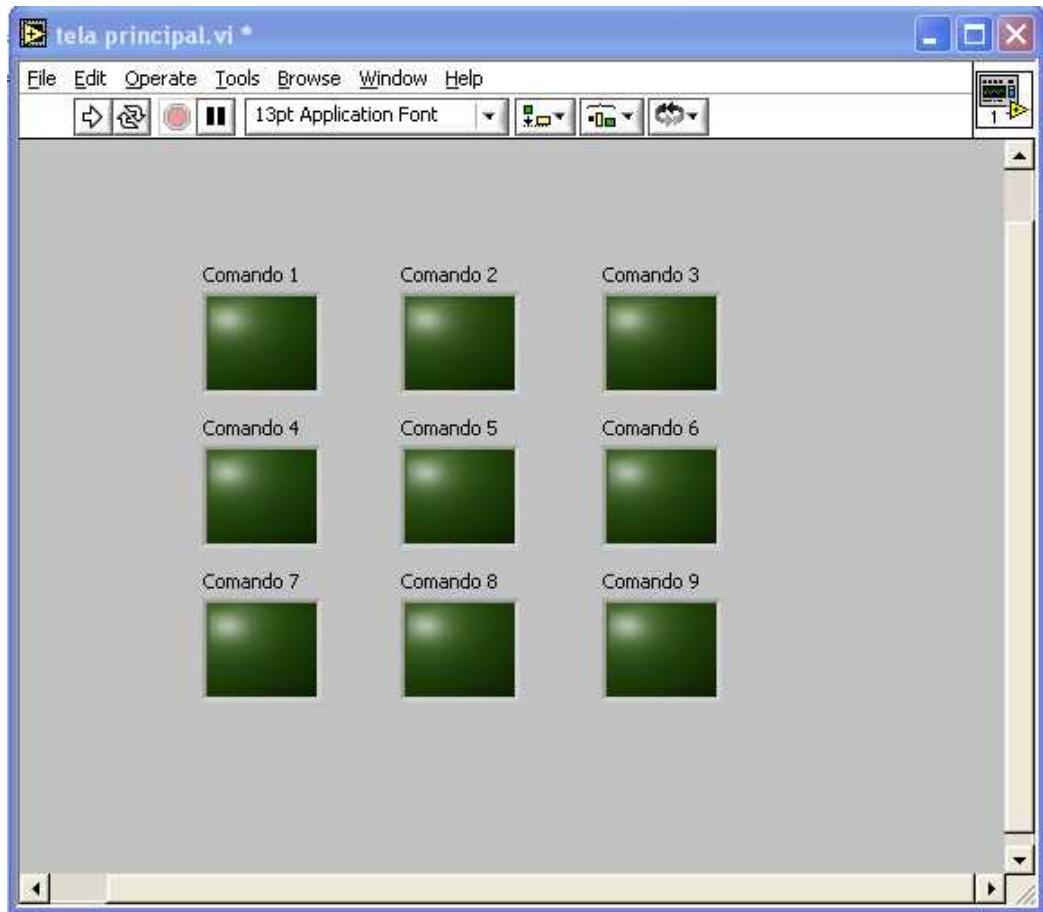


Figura 35 - Tela principal do programa implementado em LabVIEW.

Levando-se em conta a velocidade de movimentação do globo ocular, a quantidade de quadros a serem analisados depende da velocidade de aquisição e de processamento da imagem pelo computador. Uma grande quantidade de imagens é desejada, porém deve-se achar o limite mínimo para essa análise sob pena de não ser possível a análise correta.

Outro fator que também influencia o sistema são os movimentos das pálpebras, que podem ser interpretados como acionamento dos botões. Para evitar esses tipos de erros de interpretação foram estabelecidas as seguintes regras:

- O movimento sempre começa da região de repouso;
- O mínimo de quatro imagens devem ser analisadas para que haja a decisão.

- O acionamento de um botão só é possível com sua prévia seleção; o indivíduo deve ficar com o olho na posição central e fechar os olhos como em uma “piscada” de no mínimo dois segundos. Após isso, haverá um sinal sonoro e só então o sistema executará o comando associado ao botão.

- Os movimentos permitidos são sempre para os botões adjacentes horizontais e verticais. Os movimentos nas diagonais requerem outra abordagem por parte do controlador, mais complexa para implementar, podendo ocasionar um maior número de conflitos nas tomadas de decisão.

A seqüência de comandos e movimentos que devem ser utilizados são descritos no algoritmo abaixo, na figura 36.

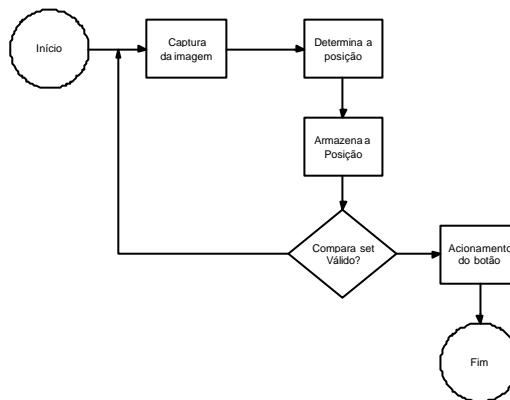


Figura 36 - Algoritmo para acionamento dos botões.

Após a tomada de decisão e do evento, o sistema volta à condição normal de operação. Na tela de interface, o início é sempre no botão central, o Comando 5 na figura 37. Para não haver dúvida sobre qual botão está selecionado, os demais ficam desabilitados como podemos ver na figura 37.

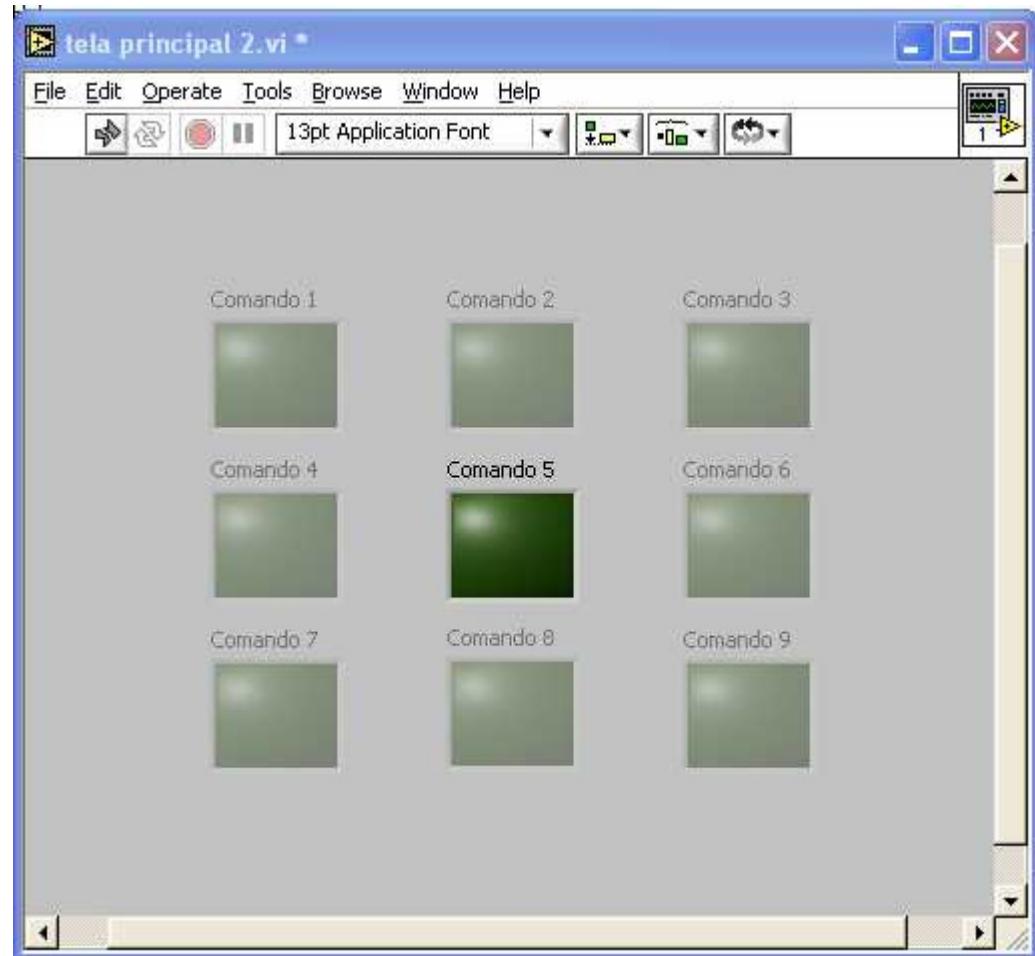


Figura 37 - Tela principal com botões desabilitados implementado em LabVIEW.

3 - Resultados

3.1 - Implementação dos Filtros

Os parâmetros dos filtros implementados foram determinados de tal forma a isolar na imagem processada somente a região da pupila. Para isso foram adotados os seguintes parâmetros:

- Filtro de potência – fator 1,5, *default* do *software*.
- Erosão da imagem – foi utilizada uma estrutura 5x5 pixels com valor de 1 para todos os pixels e com uma interação.
- Filtro gaussiano – implementado um kernel de 3x3 pixels com os seguintes pesos : primeira linha { 1 2 1 }, segunda linha { 2 4 2 } e terceira linha {1 2 1 }. Para exemplificar a necessidade desses filtros, foi colocada uma linha sobre a região da pupila para analisar a variação da intensidade de cinza (linha verde na figura 38).

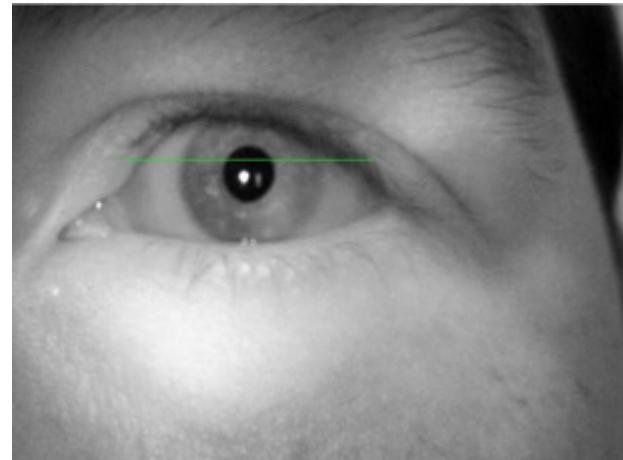


Figura 38 - Linha para análise do nível de cinza sem reflexo.

Observa-se nas figuras 39, 40, 41 e 42 a variação da intensidade de cinza dos pixels pertencentes a uma mesma linha da imagem da pupila, variando-se somente os filtros aplicados.

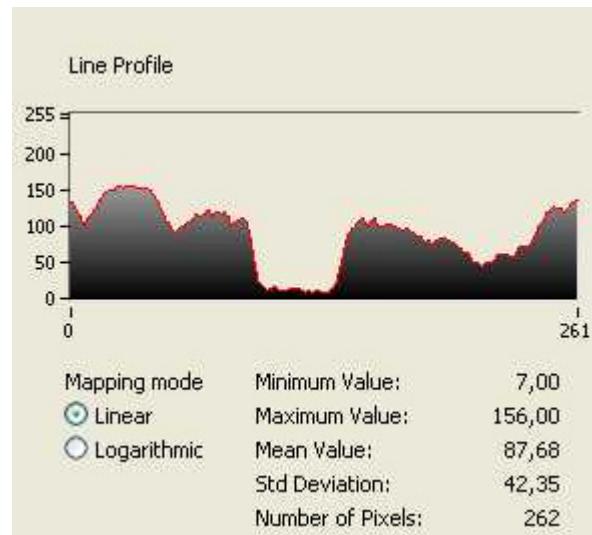


Figura 39 - Nível de cinza da imagem original.

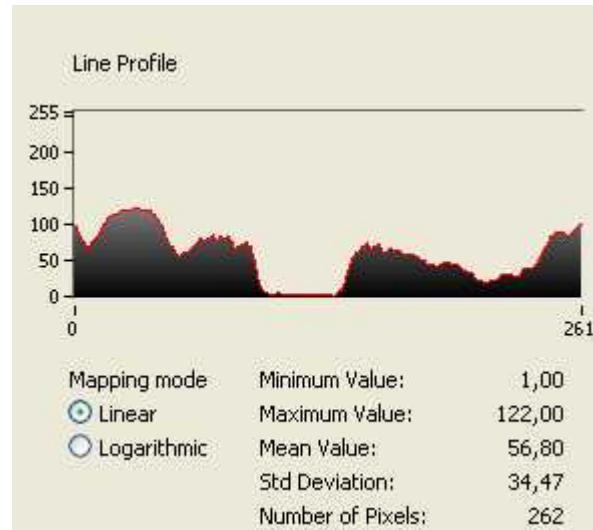


Figura 40 - Nível de cinza depois do filtro de potência.

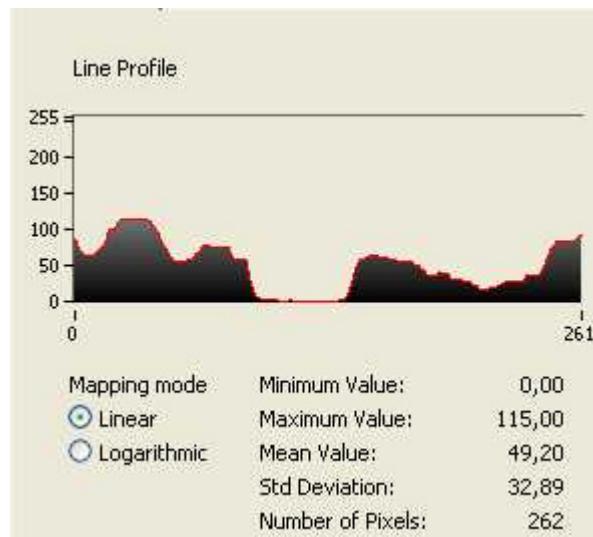


Figura 41 - Nível de cinza depois da erosão.

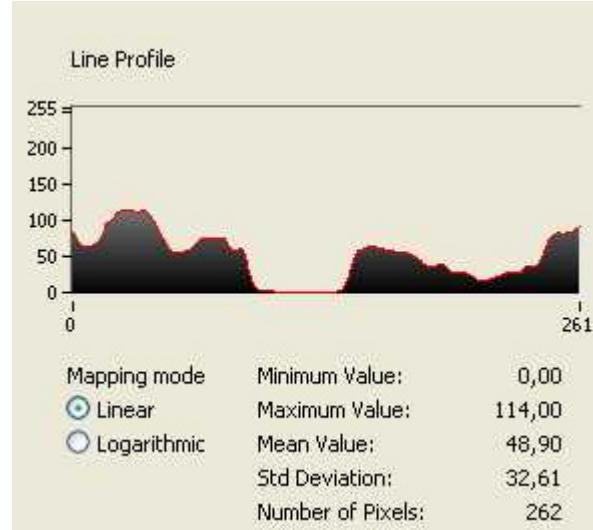


Figura 42 - Nível de cinza depois do filtro gaussiano.

Outra linha da mesma dimensão posicionada na mesma região só que passando pelo reflexo no cristalino é mostrada na figura 43. Nota-se uma atenuação do nível de cinza na região do reflexo, como mostrado nas figuras 44, 45, 46 e 47.

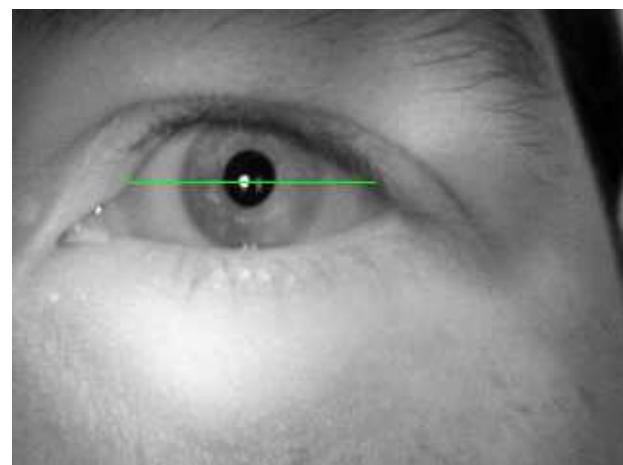


Figura 43 - Linha para análise do nível de cinza sobre reflexo no cristalino.

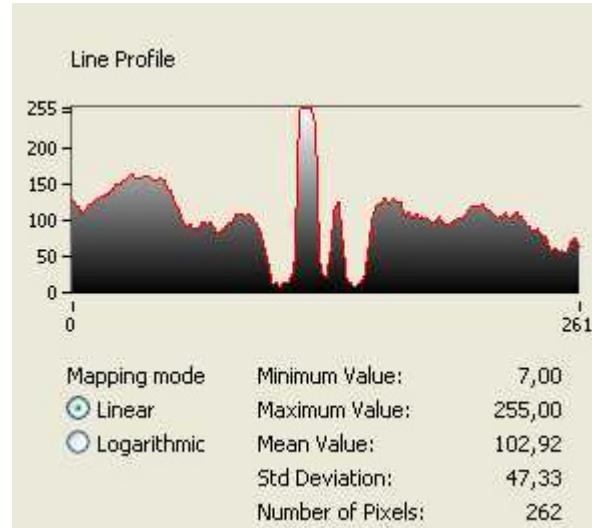


Figura 44 - Nível de cinza da imagem original com reflexo do cristalino.

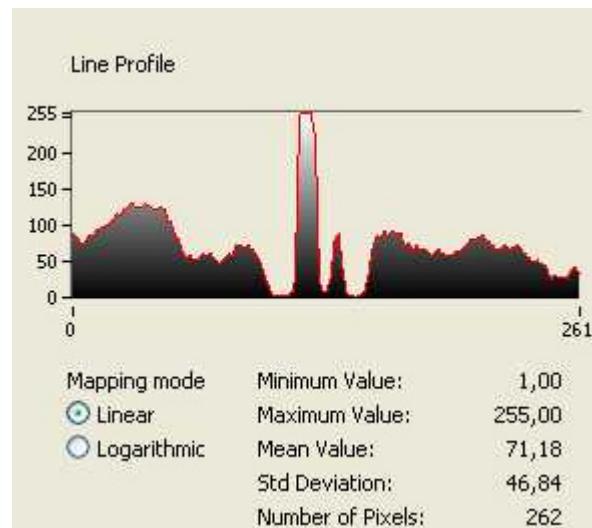


Figura 45 - Nível de cinza depois do filtro de potência.

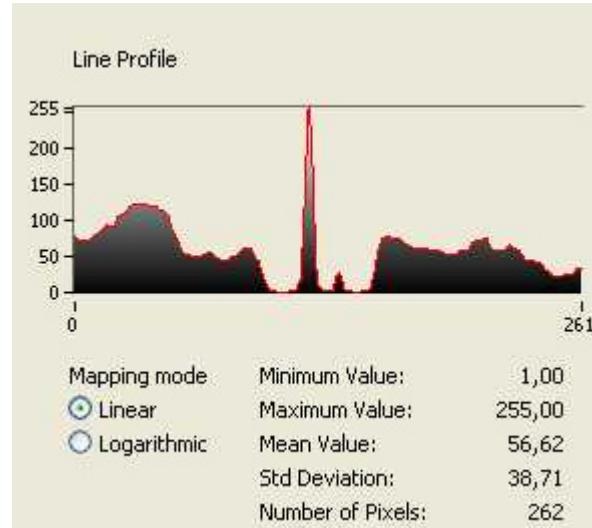


Figura 46 - Nível de cinza depois da erosão.

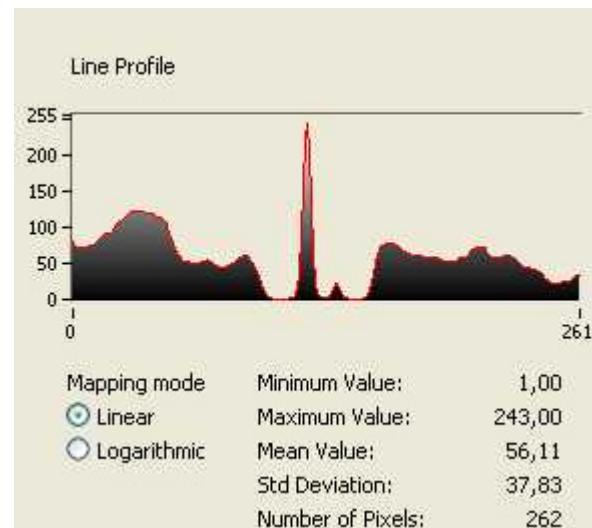


Figura 47 - Nível de cinza depois do filtro gaussiano.

- Filtro de cinza – Observando o espectro na figura 48 e sua imagem correspondente na figura 49 escolheu-se um filtro de cinza de tal maneira que a região da pupila fosse representada da maneira mais homogênea possível.



Figura 48 - Espectro de cinza da imagem.



Figura 49 - Imagem correspondente ao espectro de cinza.

Neste filtro um aumento na banda de nível de cinza, aumentando a homogeneidade da região da pupila, acarreta também no aumento de pequenos objetos (ou pontos) e de regiões nas bordas que serão selecionadas. Com essa ponderação, ajustamos a banda do filtro entre 0 e 21 em uma escala de 0 a 255 (oito bits). Exemplificando, temos as figuras 50, com a banda entre 0 e 21 e a figura 51, com a banda entre 0 e 47.

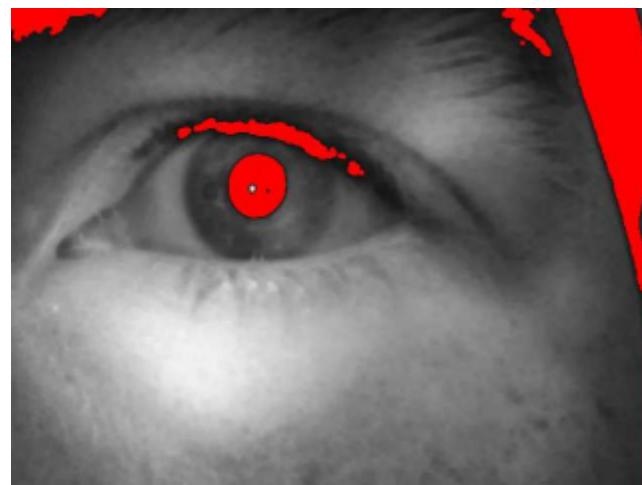


Figura 50 - Nível de cinza selecionado entre 0 e 21.

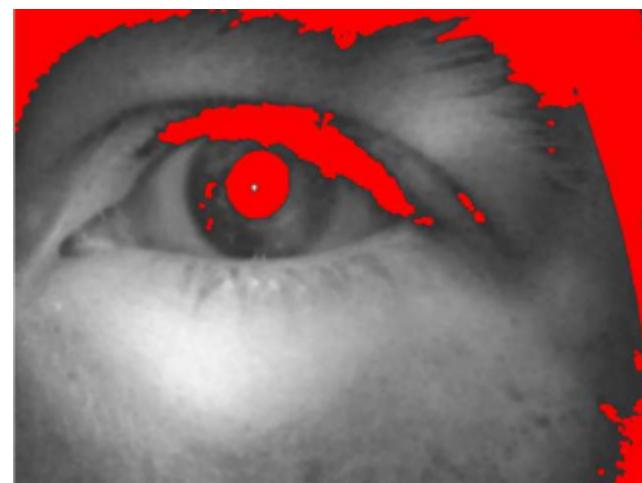


Figura 51 - Nível de cinza selecionado entre 0 e 47.

Outra característica que é afetada por este filtro é um ponto de alta luminosidade dentro da região da pupila. Esse ponto é reflexo da iluminação externa no cristalino do olho do indivíduo.

Observa-se nas superfícies mostradas nas figuras 52 (imagem sem tratamento) e 53 (imagem após o filtro gaussiano) a diminuição da área de reflexão.



Figura 52 - Espectro da imagem original.



Figura 53 - Espectro da imagem após filtro gaussiano.

Ajustando a banda do nível de cinza de tal modo que se englobe totalmente esse reflexo dentro da área da pupila, conseguimos com o filtro de preenchimento eliminar esse reflexo da imagem binarizada.

- Eliminação das bordas – Neste filtro o único parâmetro que pode ser ajustado é a conectividade, que pode ser 4 (os pixels envolvidos são os que estão dispostos horizontalmente e verticalmente em relação ao pixel que está sendo transformado) ou 8 (além dos citados em conectividade 4 envolve também os dispostos diagonalmente ao pixel de referência). Neste sistema, a mudança de conectividade não influí significamente na eliminação das bordas.

- Eliminação dos pequenos pontos – Neste filtro, além da conectividade, explicada no filtro de eliminação de bordas, pode-se ajustar o número de interações do filtro. Foi ajustado para cinco interações, eliminando assim os pequenos pontos. Um aumento no número de interações aumenta o tamanho mínimo da região a ser filtrada.

- Filtro de preenchimento – Como no filtro de eliminação de bordas, o único parâmetro ajustável é a conectividade e também não influí significamente no preenchimento.

- Localização das áreas circulares – A câmera foi posicionada a aproximadamente cinco centímetros do globo ocular e ajustado o seu foco. A partir desta distância, ajustou-se o filtro para localizar círculos com o raio mínimo de onze pixels garantindo assim, em conjunto com o filtro de nível de cinza que somente na área da pupila seja inscrito o círculo. Esse filtro tem como saída o número de círculos encontrados, seus raios e as posições de seus centros respectivamente.

- Calculo do Centróide – Esse filtro calcula o centróide da imagem em relação às áreas binarizadas. Para o cálculo ele atribui peso 1 para os pixels selecionados e peso 0 para os pixels não selecionados. Salienta-se que uma imagem com muito ruído (regiões dentro da banda de nível de cinza mas fora da área da pupila) pode levar a um resultado que não satisfaz a especificação do sistema, ou seja, a posição do centróide pode estar

fora da região da pupila. Perto da região do globo ocular a maior fonte de erro é a sobrancelha que tem valores de tons de cinza na mesma faixa da pupila.

Após os testes com o algoritmo original, procurou-se uma otimização no processamento das imagens de modo a reduzir o tempo para análise de cada imagem adquirida. O resultado foi a simplificação do algoritmo inicial, aplicando-se uma máscara na região de interesse, ou seja, a seleção de uma área próxima da área de interesse, diminuindo o tamanho da imagem a ser processada. Adotando-se esse procedimento o filtro de potência, a erosão da imagem, o filtro gaussiano, a eliminação das bordas da imagem e pequenos objetos não são mais necessários, ficando o algoritmo reduzido à:

- Aplicação da máscara na área de interesse ou seja, a região da imagem fora da máscara é descartada pelo processamento
- Filtro de cinza
- Preenchimento de buracos
- Cálculo do Centróide

A figura 54 mostra a imagem adquirida e a figura 55 mostra a imagem após a aplicação da máscara.



Figura 54 – Imagem adquirida

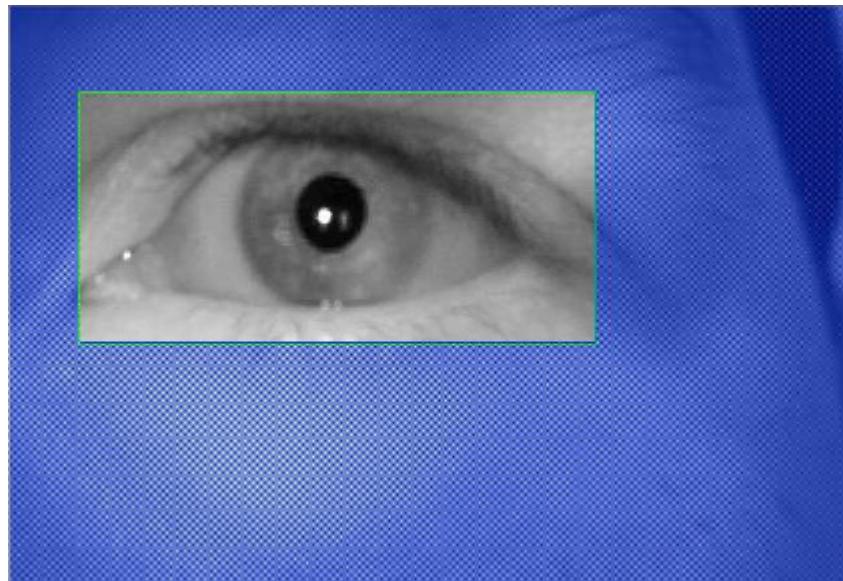


Figura 55 – Imagem após a aplicação da máscara

Um outro fator pesquisado posteriormente foi a possibilidade de utilização de uma *webcam* para a aquisição da imagem. A escolha do modelo da *webcam* levou em consideração a compatibilidade com o *software* LabVIEW, sendo escolhido o modelo *QuickCam Pro 4000* da Logitech. Este modelo captura imagens com 640x480 pixels, possui interface USB e imagem de saída colorida no padrão RGB (*Red, Green and Blue*). Neste padrão as imagens são formadas por 24 bits sendo que cada plano (vermelho, verde e azul) possui 8 bits. O processamento das imagens pelos filtros do algoritmo é realizado em 8 bits. Para não haver conflito entre a imagem adquirida e o processamento, separa-se o componente vermelho da imagem original através de um filtro próprio e utiliza-se a resposta deste filtro para o processamento da imagem. Na figura 56 observa-se a imagem adquirida pela *webcam* e na figura 57 a sua componente vermelha.



Figura 56 - Imagem adquirida pela *webcam*



Figura 57 – Plano vermelho da imagem adquirida pela *webcam*

3.2 - Tempos de Processamento

No sistema podemos dividir o tempo de processamento em:

- Tempo de aquisição – Neste caso, como a velocidade de aquisição é de 30 fps (*frames per second* – quadros por segundo), não é fator limitante para o desempenho do sistema.

- Tempo de processamento da imagem – Está dividido em função dos filtros aplicados em cada algoritmo e o método de aquisição . Os tempos encontrados para cada filtro foram estimados pelo software LabVIEW, através de função interna do software, sendo usado como plataforma de hardware um Pentium IV de 2.4GHz e 512 Mb de memória RAM, utilizando o sistema operacional WindowsXP.

Tabela 2 - Tempos de processamento com aquisição analógica

Filtro	Tempo (ms)
Filtro de Potência	1,0
Erosão	32,0
Filtro Gaussiano	16,3
Nível de cinza	<0,5
Eliminação de Bordas	2,3
Eliminação de Pequenos Objetos	8,0
Filtro de Preenchimento	4,7
Detecção de Círculo	68,7
Cálculo do Centróide	1,0

Tabela 3 - Tempos de processamento com aquisição analógica e algoritmo otimizado

Filtro	Tempo (ms)
Máscara	1,816
Nível de cinza	0,138
Filtro de Preenchimento	0,619
Detecção de Círculo	11,274
Cálculo do Centróide	0,421

Tabela 4 - Tempos de processamento com aquisição via *webcam*
(resolução de imagem de 640x480)

Filtro	Tempo (ms)
Separação Plano Vermelho	1,751
Filtro de Potência	1,096
Erosão	25,401
Filtro Gaussiano	13,578
Nível de cinza	0,448
Eliminação de Bordas	1,253
Eliminação de Pequenos Objetos	8,339
Filtro de Preenchimento	3,215
Detecção de Círculo	60,346
Cálculo do Centróide	1,018

Tabela 5 - Tempos de processamento com aquisição via *webcam* e algoritmo otimizado (resolução de imagem de 640x480)

Filtro	Tempo (ms)
Separação Plano Vermelho	1,779
Máscara	1,796
Nível de cinza	0,107
Filtro de Preenchimento	0,377
Detecção de Círculo	8,209
Cálculo do Centróide	0,327

Tabela 6 - Tempos de processamento com aquisição via *webcam*
(resolução de imagem de 320x240)

Filtro	Tempo (ms)
Separação Plano Vermelho	0,492
Filtro de Potência	0,342
Erosão	7,148
Filtro Gaussiano	2,818
Nível de cinza	0,179
Eliminação de Bordas	0,382
Eliminação de Pequenos Objetos	2,055
Filtro de Preenchimento	0,627
Detecção de Círculo	15,126
Cálculo do Centróide	0,402

Tabela 7 - Tempos de processamento com aquisição via *webcam* e algoritmo otimizado (resolução de imagem de 320x240)

Filtro	Tempo (ms)
Separação Plano Vermelho	0,463
Máscara	0,526
Nível de cinza	0,091
Filtro de Preenchimento	0,210
Detecção de Círculo	2,602
Cálculo do Centróide	0,217

Ao analisar os tempos de cada filtro, chega-se a conclusão que os filtros críticos, no fator tempo, são a detecção dos círculos e o filtro de erosão. Mesmo mudando os parâmetros dos diversos filtros não há um ganho significativo na velocidade de processamento.

- Tempo de processamento do controlador *fuzzy* – Esse tempo é função da quantidade de regras programadas no controlador. No sistema existem quatro variáveis

de entrada que são as coordenadas X e Y do Centróide, cada uma com três estados e as variáveis booleanas $X > Y$ e $X + Y > N + 1$, cada uma com dois estados, porém o editor de regras *fuzzy* determina que cada variável tenha no mínimo 3 estados, mesmo que o estado extra tenha largura mínima. Como consequência tem-se a possibilidade de 81 combinações diferentes de entradas. A saída do controlador possui somente 5 variáveis, que são Esquerda, Centro, Direita, Alto e Baixo. Algumas das combinações não são coerentes e outras são redundantes, possibilitando assim a redução do número de combinações efetivas no controlador *fuzzy*. Neste sistema, o número de combinações efetivas são 62. A figura 58 mostra o editor de regras *fuzzy*.

Rulebase - Editor									
Rule-Nr.	Coordenad	Coordenad	X>Y	X+Y>N+1	THEN	With	Up	Defuzzification Method	
1	Esquerda	Alto	Não	Não	Alto	DoS		Center of Gravity	
2	Esquerda	Alto	Não	Indiferente	None	1,00		default term	
3	Esquerda	Alto	Não	Sim	Alto	1,00		Centro	
4	Esquerda	Alto	Indiferente	Não	None	1,00		if no rule is activ	
5	Esquerda	Alto	Indiferente	Indiferente	None	1,00		Take last value	
6	Esquerda	Alto	Indiferente	Sim	None	1,00		Inference Method	
7	Esquerda	Alto	Sim	Não	Esquerda	1,00		Max-Min	
8	Esquerda	Alto	Sim	Indiferente	None	1,00		Select Form of Rulebase	
9	Esquerda	Alto	Sim	Sim	Esquerda	1,00		normal Rulebase	
10	Esquerda	Centro	Não	Não	Esquerda	1,00		total rules	81
11	Esquerda	Centro	Não	Indiferente	None	1,00		used rules	62
12	Esquerda	Centro	Não	Sim	Esquerda	1,00		default DoS	1,00
13	Esquerda	Centro	Indiferente	Não	None	1,00		Help	
14	Esquerda	Centro	Indiferente	Indiferente	None	1,00		Quit	
15	Esquerda	Centro	Indiferente	Sim	None	1,00			Dn

Figura 58 - Editor de regras fuzzy - Tela do *Fuzzy Logic Controller Design*, aplicativo pertencente ao *software LabVIEW*

O tempo gasto pelo processamento do controlador *fuzzy* não foi medido diretamente, devido à impossibilidade de isolar do sistema o controlador e medir o seu tempo de funcionamento a cada análise. Entretanto, o tempo gasto para sua execução pode ser estimado. Essa estimativa de tempo é obtida executando-se somente a parte do *software* que contém o controlador *fuzzy* e dois marcadores de tempo, um no início e outro no fim da rotina. Verificou-se que é incomensurável o tempo do controlador.

4 - DISCUSSÃO

Os testes demonstraram a possibilidade de melhoramento do desempenho do sistema utilizando os algoritmos otimizados. Neste quesito temos a comparação na tabela 8:

Tabela 8 – Tempos de Processamento dos diversos algoritmos e sistemas de aquisição

Algoritmo	Tempo (ms)	Quadros por segundo
Aquisição analógica, algoritmo completo	134,00	7,46
Aquisição analógica, algoritmo otimizado	14,85	67,34
<i>WebCam</i> , resolução 640x480, algoritmo completo	117,00	8,57
<i>WebCam</i> , resolução 640x480, algoritmo otimizado	12,60	79,36
<i>WebCam</i> , resolução 320x240, algoritmo completo	30,00	33,82
<i>WebCam</i> , resolução 320x240, algoritmo otimizado	4,00	243,21

Nota-se que a utilização da máscara diminui sensivelmente o tempo de processamento, devido à redução do número de pixels que serão processados. O fator resolução de imagem, variável apresentada na aquisição com *webcam*, mostra que não há a necessidade de uma grande resolução de imagem para tornar o sistema eficiente.

Convém salientar que a *webcam* utilizada possui na sua estrutura um filtro de infravermelho, impossibilitando a aquisição nesse espectro de freqüências. Neste caso, os parâmetros dos filtros de cinza sofrem um ajuste para selecionar a área da íris e não somente a pupila. Esse fato diminui a precisão do sistema mas não influencia de forma significativa o resultado final do processamento da imagem, vistos os resultados obtidos. Mesmo quando se aplica a máscara, reduzindo a área de processamento, a íris estará sempre nessa região. Nesse caso, a utilização de iluminação externa do ambiente também é compensada pelo ajuste nos parâmetros dos filtros.

O conjunto sistema de aquisição e processamento de imagem possui como característica principal a sua versatilidade, visto que dois sistemas distintos de aquisição de imagens possuem resultados próximos para o mesmo processamento e resolução das imagens, levando-se em conta também as peculiaridades das imagens adquiridas (monocromáticas ou coloridas).

5 - Conclusão

A proposta deste trabalho foi desenvolver um sistema de reabilitação de baixo custo para indivíduos com sérias restrições de movimento e comunicação baseado na tecnologia de uso corrente. O desenvolvimento do *software* para o sistema proposto, que no caso é uma interface homem-máquina, apresentou bons resultados na análise da imagem para a obtenção da variável de controle no que tange a questão de tempo de execução. Observa-se que o maior gasto de tempo na rotina é para a seqüência de detecção dos círculos inscritos. Outras formas de determiná-los que consomem menos recursos de processamento, reduziram de forma sensível o tempo de processamento

Embora o *hardware* do sistema esteja de acordo com a proposta do trabalho, o valor da placa de aquisição é um fator que torna o custo final do sistema não acessível à grande maioria da população. A solução utilizada na neste trabalho é o uso de câmeras digitais com saída USB (*webcams*), que são de custo relativamente baixo e facilmente encontradas no comércio.

A portabilidade do sistema utilizando *webcams* possibilita a sua aplicação somente com um computador, sem a necessidade de *hardware* específico. A evolução deste sistema será a implementação com câmeras com saída *firewire* (protocolo IEEE 1394), visto que possuem melhor resolução que as *webcams*, sua conexão com o computador requer apenas uma conexão *firewire* e diversos modelos no mercado adquirem tanto o espectro visível como o de infravermelho.

Os parâmetros utilizados no processamento das imagens permitem ao sistema uma adaptação a ambientes com diversos tipos de iluminação, sendo assim desnecessária a adaptação do ambiente ao sistema.

A facilidade de implementação do controlador *fuzzy* permite que as mais diversas configurações das áreas de atuação do controlador possam ser ajustadas para diversos indivíduos de maneira simples e rápida.

A flexibilidade do sistema até aqui desenvolvido permite sua fácil adaptação a uma ampla gama de *softwares* e *hardwares* para controle de sistemas eletro-eletrônicos do cotidiano de um indivíduo, tais como televisão, rádios, luzes, ventiladores, alarmes e afins, possibilitando a integração deste indivíduo ao seu ambiente.

BIBLIOGRAFIA

- ARZI, M.; MAGNIN, M. (1989). A fuzzy set theoretical approach to automatic analysis of nystagmic eye movement. *IEEE Transactions on Biomedical Engineering*, New York, v.36, n 9,p.954-963, Sept.
- BAXTER, R. (2007). Laser safety training manual. University of Chicago disponível em :<<http://chemistry.uchicago.edu/safety/LaserSafety.pdf>>. Acesso em 10 Feb. 2007.
- CHEN, Y.L.; TANG, F.T. et al. (1999). The new design of an infrared-controlled human-computer interface for the disabled. *IEEE Transactions on Rehabilitation Engineering*, New York, v.7, n.4, p.474-481, Dec.
- CAMERAMOUSE (2006). Disponível Em:<<http://www.cameramouse.com/>>. Acesso em: 15 Feb. 2006.
- EVANS, D.G.; DREW, R.; BLENKHORN, P. (2000). Controlling mouse pointer position using an infrared head-operated joystick. *IEEE Transactions on Rehabilitation Engineering*, New York, v.8, n.1, p.107-117, Mar.
- GONZALEZ, R.C.; WOODS, R.E. (1993). *Digital image processing*. Reading, MA: Addison-Wesley.
- HORI, J.; SAKANO, K.; SAITO, Y.; (2004). Development of communication supporting device controlled by eye movements and voluntary eye blink. In: ENGINEERING IN MEDICINE AND BIOLOGY SOCIETY, 2004. *Proceedings...* v.6, p.4302-4305.
- KAUFMAN, A.E.; BANDOPADHAY, A.; SHAVIV, B.D. *An eye tracking computer user interface*. Disponível em:<<Http://www.cs.sunysb.edu/~vislab/projects/eye/index.html>>. Acesso em: 10 Jan. 2006.
- KAUFMAN,A.E.; BANDOPADHAY,A.; SHAVIV, B.D. (1993). An eye tracking computer user interface. In: RESEARCH FRONTIERS IN VIRTUAL REALITY WORKSHOP, 1993. *Proceedings...* New York: IEEE Computer Society. p.120-121.

- LACOURSE, J.R; HLUDIK JR., F.C. (1990). An eye movement communication control system for the disabled. *IEEE Transactions on Biomedical Engineering*, New York, v.37, n.12, p.1215-1220, Dec.
- NATIONAL INSTRUMENTS (2000a). *LabVIEW User's Manual* Texas.
- _____. (2000b). *IMAQ vision concepts*. Texas.
- PROJETO Eagleeyes. Boston: Boston College. Disponível em:<<http://www.bc.edu/schools/csom/eagleeyes/eagleeyes/gallery/page9/>>. Acesso em: 15 Fev. 2006.
- SHAW, I.S.; SIMÕES, M.G. (1999). *Controle e modelagem fuzzy*. São Paulo: Edgard Blücher.
- ATLAS interativo da anatomia humana. (1999). Novartis Medical Education, Artmed.
- TSOUKALAS, L.H.; UHRIG, R.E. (1997). *Fuzzy and neural approaches in engineering*. New York: John Wiley.
- VAS, P. (1999). *Artificial-intelligence-based electrical machines and drives – application of fuzzy, neural, fuzzy-neural, and genetic-algorithm-based techniques*. Oxford: Oxford University Press.
- WIJESOMA, W.S.; KANG SAY WEE et al.(2005). EOG based control of mobile assistive platforms for the severely disable. *IEEE International Conference on Robotics and Biomimetics*. Shatin p.490-494.
- YAGER, R.R.; FILEV, D.P. (1994). *Essentials of fuzzy modeling and control*. New York: John Wiley.
- YINGXI CHEN; NEWMAN, W.S. (2004) A human-robot interface based on electrooculography. In: ROBOTICS AND AUTOMATATION, 2004. *Proceedings...* v.1, p.243-248.

7 - Apêndice⁽¹⁾

Definição de kernel gaussiano

O filtro de convolução gaussiano é um filtro de média e seu kernel é baseado no seguinte modelo

$$\begin{array}{ccc} a & d & c \\ b & x & b \\ c & d & a \end{array}$$

onde a, b, c e d são números inteiros e $x > 1$.

Sendo todos os coeficientes de um kernel gaussiano positivos, cada pixel é resultado da média ponderada de seus vizinhos. Quanto maior o peso de um pixel adjacente, maior a sua influência no novo valor calculado para o pixel central.

O coeficiente central de um filtro gaussiano é maior que 1. Portanto o valor original do pixel é multiplicado por um peso maior que o peso de qualquer um de seus adjacentes.

Como resultado, um coeficiente central maior corresponde a um efeito de suavização lenta. Um kernel de maior dimensão corresponde a um efeito de suavização mais abrupta.

A seguir, uma tabela com os kernels gaussianos pré-definidos no software LabVIEW, no módulo VisionBuilder.

3x3

$$\begin{array}{ccccccccc} 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 & 4 & 1 & 1 & 2 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{array} \quad \begin{array}{ccccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 1 \\ 1 & 4 & 1 & 4 & 1 & 4 & 1 & 4 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 1 \end{array} \quad \begin{array}{ccccccccc} 1 & 2 & 1 & 2 & 4 & 2 & 4 & 16 & 4 \\ 4 & 16 & 4 & 16 & 64 & 64 & 64 & 256 & 256 \\ 1 & 4 & 1 & 4 & 16 & 16 & 16 & 64 & 64 \end{array}$$

5x5

1	2	4	2	1
2	4	8	4	2
4	8	16	8	4
2	4	8	4	2
1	2	4	2	1

7x7

1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1