

Daniel Rodrigues Carvalho

# **Reconhecimento e Rastreamento da Direção do Olhar Para Auxílio de Deficientes Físicos**

Brasil

2015, v-1.0



Daniel Rodrigues Carvalho

## **Reconhecimento e Rastreamento da Direção do Olhar Para Auxílio de Deficientes Físicos**

Monografia apresentada ao curso de bacharelado em Engenharia de Computação da Universidade Federal da Bahia como requisito parcial para a obtenção de grau de bacharel em Engenharia de Computação.

Universidade Federal da Bahia - UFBA  
Departamento de Engenharia de Computação  
Programa de Graduação

Orientador: Maurício Pamplona Segundo

Brasil  
2015, v-1.0

Daniel Rodrigues Carvalho

Reconhecimento e Rastreamento da Direção  
do Olhar Para Auxílio de Deficientes Físicos/ Daniel Rodrigues Carvalho. – Brasil,  
2015, v-1.0-

53 p. : il. (algumas color.) ; 30 cm.

Orientador: Maurício Pamplona Segundo

Trabalho de Conclusão de Curso – Universidade Federal da Bahia - UFBA  
Departamento de Engenharia de Computação  
Programa de Graduação, 2015, v-1.0.

1. Reconhecimento Facial. 2. Rastreamento Facial. 3. Rastreamento Ocular. I.  
Orientador: Maurício Pamplona Segundo. II. Universidade Federal da Bahia - UFBA  
Departamento de Engenharia de Computação  
Programa de Graduação. III. Reconhecimento e Rastreamento da Direção  
do Olhar Para Auxílio de Deficientes Físicos.

Daniel Rodrigues Carvalho

## **Reconhecimento e Rastreamento da Direção do Olhar Para Auxílio de Deficientes Físicos**

Monografia apresentada ao curso de bacharelado em Engenharia de Computação da Universidade Federal da Bahia como requisito parcial para a obtenção de grau de bacharel em Engenharia de Computação.

Trabalho aprovado. Brasil, 25 de julho de 2019:

---

**Maurício Pamplona Segundo**  
Orientador

---

**Rubisley de Paula Lemes**  
Convidado 1

---

**Luciano Rebouças de Oliveira**  
Convidado 2

Brasil  
2015, v-1.0



*À minha família e falta de juízo ao escolher este curso.*





# Agradecimentos

Agradeço aos professores Rubisley e Maurício por incentivarem a obtenção de conhecimento e o desafio pessoal.



*“Mesmo que não possamos entender os pensamentos das pessoas, através da combinação de movimentos oculares com outras informações, podemos entender coisas muito mais profundas, como o comportamento humano, por exemplo. Entender coisas tão profundas assim significa que computadores podem suportá-las. Isto significa que podemos criar tecnologias que não existem ainda.”*

*(Autor Desconhecido)*



# Resumo

Tendo-se em vista a dificuldade de utilização de recursos computacionais por pessoas com deficiências locomotoras, o objetivo desta monografia é criar um sistema de baixo custo para ajudá-las a utilizar computadores aplicando-se algoritmos de detecção de olho, pupila e estimativa da orientação da cabeça para emular o movimento do cursor do mouse através da direção do olhar do usuário. Através de técnicas como rastreamento facial não rígido, detecção de pupila por contraste entre esclera e pupila, e estimativa de direção do olhar baseada na distância entre a tela e o usuário, um software foi criado para simular a utilização do mouse, permitindo translação e cliques baseados na movimentação dos olhos. O sistema resultante apresenta precisão superior à de usuários utilizando equipamentos convencionais para controle do mouse, e pontos de melhoria são sugeridos ao final do estudo.

**Palavras-chave:** Rastreamento de Direção do Olhar. Detecção de Piscada.



# Abstract

With the difficulty of computer resources use by people with locomotion deficiency in sight, the objective of this paper is to create a low cost system to help them use computers by using eye and pupil detection and head orientation estimation algorithms to emulate mouse cursor movement via user's gaze direction. Through techniques as non-rigid face tracking, pupil detection by sclera-pupil contrast, and gaze estimation based on the distance between the screen and the user, a software has been created to simulate mouse utilization, allowing translation and clicks based on eye movement. The resulting system presents accuracy higher than user using conventional equipment for mouse control, and improvable points are suggested at the end of the study.

**Keywords:** Gaze Tracking. Blink Detection.





# Lista de ilustrações

Figura 1 – Visão geral do sistema. A cada iteração do sistema, uma imagem do usuário é recebida e processada, gerando a estimativa de posição focal.	31
Figura 2 – Fluxograma dos modos de operação. A partir de um comando de mudança de modo de operação (piscada com 2 olhos), o sistema entra no modo Botão Esquerdo, no qual realizará magnificações na imagem até obter precisão suficiente para emitir um clique e retornar ao modo Visualização.	32
Figura 3 – Pontos detectados pelos diferentes detectores combinados.	34
Figura 4 – Pontos detectados pelos diferentes detectores combinados.	34
Figura 5 – Detecção de pupila e piscada.	36
Figura 6 – Estimativa correta de direção do olhar.	38
Figura 7 – Estimativa incorreta de direção do olhar.	39
Figura 8 – Sequência de ampliações para a realização de um clique.	41
Figura 9 – Distribuição Atual de Resoluções de Monitores.	42
Figura 10 – Comparação entre as estimativas de pontos focais com e sem filtragem. A linha tracejada representa o ponto estimado após a aplicação do filtro. A reta pontilhada representa a posição real do ponto.	43
Figura 11 – Sistema online: O usuário clica o mais próximo do centro do alvo para gerar a estimativa de precisão.	45
Figura 12 – Histogramas das precisões alcançadas nos experimentos.	46
Figura 13 – Detecção de fechamento de olhos. Um quadrado verde é mostrado no lado no qual a piscada é detectada.	47



# Lista de abreviaturas e siglas

AAM	Active Appearance Models
FIFO	First In, First Out
HOG	Histogram of Oriented Gradients
OpenCV	Open Source Computer Vision
POSIT	Pose from Orthography and Scaling with Iterations



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>21</b>
<b>1.1</b>	<b>Introdução</b>	<b>21</b>
<b>2</b>	<b>TRABALHOS RELACIONADOS</b>	<b>23</b>
<b>2.1</b>	<b>Métodos Monoculares</b>	<b>23</b>
2.1.1	Câmeras USB fixas	23
2.1.2	Câmeras Móveis	23
2.1.3	Câmeras em Infravermelho	24
<b>2.2</b>	<b>Múltiplas Câmeras</b>	<b>24</b>
<b>2.3</b>	<b>Método Proposto</b>	<b>24</b>
<b>3</b>	<b>ASPECTOS TEÓRICOS</b>	<b>25</b>
<b>3.1</b>	<b>Rastreamento Facial e Estimativa de Pose</b>	<b>25</b>
<b>3.2</b>	<b>Detecção de Pupila</b>	<b>27</b>
3.2.1	Métodos Baseados em Formas	27
3.2.2	Métodos Baseados em Aparência	28
3.2.3	Métodos Híbridos	28
3.2.4	Sensor em Infravermelho	28
<b>3.3</b>	<b>Estimativa do Ponto Focal</b>	<b>28</b>
<b>4</b>	<b>O SISTEMA</b>	<b>31</b>
<b>4.1</b>	<b>Modos de Operação</b>	<b>31</b>
4.1.1	Visualização	32
4.1.2	Botão Esquerdo	32
<b>4.2</b>	<b>Cálculo da Direção do Olhar</b>	<b>33</b>
4.2.1	Rastreamento Facial e Estimativa de Pose	33
4.2.2	Detecção de Pupilas	33
4.2.3	Estimativa da Distância da Tela	36
4.2.4	Estimativa do Ponto Focal	36
<b>4.3</b>	<b>Detecção de Piscada</b>	<b>38</b>
<b>4.4</b>	<b>Ajuste Fino</b>	<b>40</b>
<b>4.5</b>	<b>Filtragem</b>	<b>42</b>
<b>5</b>	<b>RESULTADOS</b>	<b>45</b>
<b>6</b>	<b>CONCLUSÃO E TRABALHOS FUTUROS</b>	<b>49</b>

**REFERÊNCIAS** ..... 51

# 1 Introdução

## 1.1 Introdução

A deficiência locomotora é uma limitação do funcionamento do sistema músculo-esquelético, que pode ser parcial ou total. Geralmente é causada por fatores genéticos, infecções virais ou bacterianas, ou traumas. Esta limitação pode dificultar a realização de atividades diárias, como a utilização de tecnologias, alimentação e prática de esportes. Com o objetivo de facilitar a utilização de computadores por pessoas com deficiências, é proposta a criação de um software emulador de movimentos do cursor do mouse controlado pela direção do olhar e pelo piscar dos olhos, tornando possível a inserção delas nos meios tecnológico e informacional.

Trabalhos similares já foram desenvolvidos, mas a maioria deles ou necessitam que o usuário se posicione estaticamente (*i.e.* não é permitida rotação da cabeça), como é o caso do OpenGazer ([ZIELINSKI,](#) ), ou são significativamente caros devido ao uso de hardware especializado ([TOBII, 2010](#))([FUJITSU, 2012](#)). Além disso, eles normalmente não conseguem realizar tarefas que necessitam de alta precisão devido à existência de erros de cálculos causados pela falta de exatidão dos detectores de características ou pela baixa resolução da imagem.

O sistema desenvolvido ainda não permite uma interação humano-computador completa, já que foram apenas implementados os modos de visualização e de clique do botão esquerdo, mas ele fornece independência suficiente para realizar pequenas tarefas, como navegar na web e assistir vídeos. Esta monografia propõe uma maneira de aumentar consideravelmente a precisão do ponto focal, ao custo do tempo de convergência. O método proposto não é computacionalmente custoso e pode ser implementado em qualquer sistema similar.

Esta monografia está dividida da seguinte forma: o Capítulo 2 apresenta trabalhos relacionados; No Capítulo 3 os aspectos teóricos do sistema são descritos, tais como métodos de detecção e rastreamento de pontos de interesse na região facial (características faciais), detecção de íris/pupila, algoritmos para estimativa de pose (orientação da cabeça), e estimativa de ponto focal; O Capítulo 4 descreve o sistema e os aspectos de engenharia aplicados para viabilizar o funcionamento preciso do mesmo; O Capítulo 5 apresenta os resultados da simulação; No Capítulo 6 conclui-se o estudo e são apresentadas sugestões de melhorias para trabalhos futuros.





## 2 Trabalhos Relacionados

Trabalhos similares já foram desenvolvidos e comercializados, e podem ser classificados quanto ao número de câmeras em duas categorias: monoculares e múltiplas câmeras. Este capítulo discorrerá sobre estas categorias, explicitando vantagens e dificuldades.

### 2.1 Métodos Monoculares

Métodos monoculares baseiam-se na utilização de uma única câmera. Tais métodos podem ser baseados em câmeras USB fixas, móveis ou em infravermelho. Outras abordagens existem, como o rastreamento por eletrooculografia (MAJARANTA; BULLING, 2014), mas elas não serão discutidas nesta tese por serem excessivamente caras e intrusivas.

#### 2.1.1 Câmeras USB fixas

Na utilização de uma câmera USB fixa, várias restrições são criadas, como a falta de informação de distância das características faciais à câmera, além da baixa resolução da imagem. Sistemas baseados nesta abordagem normalmente aplicam algoritmos de detecção e rastreamento facial para detectar pontos de interesse no olho e estimar a orientação da cabeça (KO; YU; JEONG, 2005; VALENTI; SEBE; GEVERS, 2012; BOULAY, 2008). Apesar de imprecisos, estes métodos são baratos e fáceis de reproduzir, já que a maioria dos computadores pessoais atualmente vem com câmera embutida.

#### 2.1.2 Câmeras Móveis

Câmeras móveis são posicionadas na cabeça do utilizador e direcionadas para os olhos, e, por causa disso, os sistemas que as utilizam não necessitam de etapas de rastreamento facial e cálculo de orientação da cabeça. Tais sistemas funcionam rastreando a pupila e estimando a direção do olhar.

Técnicas como o algoritmo Stardust (LI; BABCOCK; PARKHURST, 2006), que aplica redução de ruído por filtro Gaussiano, redução da reflexão das córneas por interpolação de intensidade radial, e detecção de pupilas pelo algoritmo RANSAC (FISCHLER; BOLLES, 1981), podem ser usadas para rastreamento de pupila, devido à alta resolução da imagem adquirida do olho.

Como a câmera localiza-se a uma distância pequena e conhecida dos olhos, ela fornece maior resolução da região ocular se comparada aos outros métodos descritos nesta seção. Além disso, como os pontos faciais de interesse são facilmente detectados, a

estimativa de direção do olhar é direta. A maior desvantagem da utilização desta abordagem é a intrusividade em relação à utilização de câmeras USB fixas, já que estas câmeras são posicionadas diretamente na cabeça do usuário.

### 2.1.3 Câmeras em Infravermelho

Sistemas baseados na detecção do olhar em infravermelho são menos acessíveis, já que eles precisam tanto de emissores de luz infravermelha quanto de câmeras com leitores de radiação infravermelha. Eles funcionam através da análise da relação espacial entre a reflexão da córnea gerada pela luz infravermelha e a pupila.

Algumas técnicas já foram aplicadas para criar rastreadores comerciais da direção do olhar (TOBII, 2010; FUJITSU, 2012). O rastreador ocular de Fujitsu, por exemplo, estima a direção do olhar detectando os possíveis pontos de reflexão de córnea (círculos brancos) e candidatos a pupila (círculos pretos), aplicando em seguida regras de consistência, como a relação entre as posições da pupila, da reflexão da córnea, do emissor de luz infravermelho e da câmera, além da relação com os resultados anteriores.

## 2.2 Múltiplas Câmeras

A utilização de uma única câmera não gera informação suficiente para calcular a profundidade dos pontos e a direção do olhar precisamente. Este problema pode ser contornado se duas ou mais câmeras forem usadas, seja através da aplicação de triangulação (CHEN et al., 2008; KAWATO; TETSUTANI, 2004), pesos aos pontos possíveis (ARAR; GAO; THIRAN, 2015) ou abordagens similares nas imagens capturadas.

Estes métodos fornecem mais informação facial e geralmente geram erros menos graves. Entretanto, como estas abordagens necessitam de mais de uma câmera, estas são mais caras e precisam de calibração especial para o posicionamento das câmeras.

## 2.3 Método Proposto

Apesar dos métodos que utilizam múltiplas câmeras e métodos baseados em câmeras móveis ou em infravermelho serem mais precisos, no sistema proposto técnicas baseadas em uma câmera USB fixa são implementadas devido ao baixo custo e maior acessibilidade. O algoritmo de melhoria de precisão proposto nesta monografia pode, contudo, ser empregado em outras abordagens.

## 3 Aspectos Teóricos

Este capítulo descreve os aspectos teóricos referentes aos processos necessários para a detecção do ponto focal. São retratados métodos e algoritmos frequentemente utilizados para rastreamento facial, detecção de pupila e estimativa de pose. Além disso, é descrita a álgebra da estimativa do ponto focal.

### 3.1 Rastreamento Facial e Estimativa de Pose

O método proposto por Viola e Jones ([VIOLA; JONES, 2001](#)), primeiro algoritmo de detecção de faces em tempo real, trabalha com a detecção e seleção de características pré-determinadas, as características tipo Haar. Com ele, diversos algoritmos para rastreamento facial em tempo real puderam ser desenvolvidos e implementados, permitindo a criação de softwares para reconhecimento biométrico, vigilância por vídeo, fotografia, e direcionamento de propagandas baseado no objeto observado, por exemplo.

Os seres humanos são distintos por natureza, e, por tal motivo, apesar da similaridade, não possuem os mesmos formatos faciais. Assim, os métodos de rastreamento facial e estimativa de pose que não utilizam modelos rígidos, isto é, métodos nos quais os pontos faciais de interesse são deformados para adequar-se às faces em análise, tendem a ser mais precisos ([MURPHY-CHUTORIAN; TRIVEDI, 2009](#)). Como a maioria dos algoritmos de estimativa de orientação de cabeça utilizam técnicas para rastreamento facial, dar-se-á foco à descrição dos métodos de estimativa de pose. A estimativa precisa de pose facial é vital para o desenvolvimento do sistema, já que a orientação da cabeça possui uma relação direta com o ponto focal, fazendo com que ligeiras inclinações da cabeça alterem a posição do ponto estimado de forma significativa.

A maioria dos algoritmos de estimativa de pose pertencem a uma das oito classes a seguir ([MURPHY-CHUTORIAN; TRIVEDI, 2009](#)): métodos de modelo de aparência; vetores de detecção; métodos de regressão não linear; métodos de incorporação múltipla; modelos flexíveis; métodos geométricos; métodos de rastreamento; e métodos híbridos.

Métodos de modelo de aparência comparam imagens de um banco de dados com a imagem da pose atual para encontrar a orientação mais próxima. Apesar de serem simples e fáceis de serem treinados, algoritmos desta classe tendem a negligenciar a identidade humana, fazendo com que orientações iguais de pessoas distintas obtenham estimativas de pose diferentes.

Os vetores de detecção consistem em diversos detectores faciais, cada qual treinado para detectar faces em uma determinada pose, e o detector que tiver maior qualidade

de detecção indica a pose correta. O algoritmo proposto por Tran *et al.* (TRAN *et al.*, 2011), por exemplo, independe da captura de pontos característicos ou da aparência do modelo, permitindo a estimativa de poses mesmo quando apenas a nuca do indivíduo é visível na imagem, utilizando Histogramas de Gradientes Orientados (HOG) (DALAL; TRIGGS, 2005) para detectar a cabeça e sua direção. Apesar da grande evolução dos detectores faciais, esta classe tende a não apresentar execução em tempo real, além de possuir dificuldade na descrição dos exemplos positivos e negativos de treinamento, uma vez que os detectores devem ser mutuamente exclusivos.

Os métodos de regressão não linear realizam a estimativa através de um mapeamento não linear do espaço da imagem a uma ou mais direções de pose, em sua maioria utilizando redes neurais. As grandes vantagens destes métodos são rapidez, treinamento fácil e alta precisão. O principal problema é que eles são muito suscetíveis a erros devido à localização imprecisa da cabeça

A incorporação múltipla agrega imagens em tubos de distribuição em espaço de menor grau, isto é, para o caso da movimentação da cabeça, por exemplo, os seis graus de liberdade são interpretados em vários conjuntos de menor dimensão, *i.e.* pode-se usar um conjunto de translação tridimensional que desconsidera a rotação da cabeça para calcular o posicionamento global da face, outro conjunto considerando apenas a rotação, outro misturando a translação em um eixo e a rotação em dois eixos, e assim em diante. Cada conjunto modela variações contínuas na pose, e as imagens são projetadas neles através de técnicas lineares ou não lineares. Uma dificuldade de utilização destes métodos é a heterogeneidade humana, que dificulta o treinamento genérico das poses analisadas em cada conjunto.

Modelos flexíveis permitem a comparação de posicionamento de pontos fiduciais (pontos característicos da face) ao invés de imagens, e funcionam através da deformação de um modelo não rígido para o encaixe na face de uma imagem a ser analisada. O algoritmo descrito por Martins (MARTINS, 2008), por exemplo, consiste na combinação da utilização de Modelos de Aparência Ativos (AAM) (COOTES; EDWARDS; TAYLOR, 2001) e Pose de Ortografia e Dimensionamento com Iterações (POSIT) (DEMENTHON; DAVIS, 1995) para dedução da pose. Uma face é procurada na imagem, e AAM é aplicado para encontrar os pontos fiduciais. Deste modo, os pontos são comparados com um modelo tridimensional de uma face genérica e, através da utilização do POSIT, os ângulos de rotação da cabeça são estimados. A utilização de AAM apresenta tolerância propícia a erros da detecção da cabeça, já que o modelo se adapta à imagem para encontrar os pontos fiduciais, porém é limitada pela auto-occlusão, já que todos os pontos devem ser encontrados, o que não é possível em certas orientações da cabeça.

Métodos geométricos, assim como os modelos flexíveis, baseiam-se na detecção de pontos fiduciais, e calculam a angulação através dos ângulos entre pontos de controle,

como por exemplo, cantos da boca, cantos dos olhos e ponta do nariz. São simples, porém dependem da inexistência de oclusões, fator comum em rotações maiores de cabeça, em usuários de óculos, e em pessoas de cabelo longo. Além disso, demandam uma detecção precisa dos pontos de referência.

Os métodos de rastreamento operam através de cálculos do movimento relativo entre quadros consecutivos de vídeo, e da aplicação da discrepância à pose estimada. Apesar dos cálculos serem consideravelmente mais rápidos do que os das outras classes, eles necessitam de uma estimativa inicial precisa da pose para que o funcionamento seja adequado.

Por fim, métodos híbridos combinam características dos métodos citados anteriormente para gerar a estimativa de tal forma que as deficiências de um método sejam compensadas por outros.

## 3.2 Detecção de Pupila

A detecção de íris ou pupila é um desafio, tanto devido à oclusão gerada pela rotação da face ou pela interferência de cabelo, óculos, e outros utensílios, quanto pela própria diferença entre os olhos humanos, causada pela existência de hiperpigmentação periorbital (olheiras), flacidez excessiva das pálpebras superiores, e diferentes formatos e angulações da região. Diversas técnicas existem, podendo ser divididas em métodos baseados em formas, métodos baseados em aparência, e métodos híbridos (HANSEN; JI, 2010). Outra técnica muito utilizada é a identificação através de sensores em infravermelho.

### 3.2.1 Métodos Baseados em Formas

São métodos que identificam a região do olho através do formato. Utilizando-se uma abordagem simplificada, a região ocular pode ser descrita por elipsóides. Dentro desta geralmente empregam-se limiares e detectores de borda para isolar a região da pupila/íris.

Algumas abordagens assumem um formato circular para a pupila, e utilizam a Transformada de Hough (DUDA; HART, 1972) para encontrar círculos dentro da região ocular. Há também descritores baseados em HOG e heurísticas (TIMM; BARTH, 2011), que dependem de pontos de máximo local e conseqüentemente identificam mais facilmente falsos positivos devido à influência de sobrancelhas, óculos e ruídos na imagem. Outras técnicas são de maior complexidade, e incluem mais detalhes durante a detecção, descrevendo o olho por uma quantidade maior de parâmetros (YUILLE; HALLINAN; COHEN, 1992). Apesar da melhor qualidade de detecção, geralmente sofrem mais com erros devido a oclusões e demandam mais poder computacional.

### 3.2.2 Métodos Baseados em Aparência

Métodos baseados em aparência caracterizam-se pela comparação de características da imagem com modelos, realizando a identificação da região da pupila através da distribuição de cores/intensidades ou da resposta a determinados filtros.

Os modelos podem ser compostos por imagens contendo o posicionamento e a informação de intensidade dos pixels, como por exemplo, imagens base de diferentes olhos em diferentes posições e rotações, ou apenas compostos por dados estatísticos referentes àquela região, como por exemplo, a discrepância de intensidade de cores entre a região da esclera e da pupila. Devido à necessidade dos modelos responderem positivamente em diferentes poses faciais, o treinamento da base deve ser realizado cautelosa e extensivamente. Geralmente, estes algoritmos são computacionalmente custosos devido à realização de comparações com múltiplos elementos de alta dimensionalidade.

### 3.2.3 Métodos Híbridos

Estes métodos tentam superar as desvantagens de cada um dos métodos anteriores realizando uma abordagem mista. É possível, por exemplo, combinar informação de cor e formato, sabendo-se que a pupila geralmente tem coloração escura e um formato semicircular, e é rodeada pelo branco da esclera e pela cor da pele do usuário. Aplicando-se as heurísticas de forma ideal, pode-se reduzir o escopo de opções, e aumentar a precisão da detecção.

### 3.2.4 Sensor em Infravermelho

Técnicas que utilizam sensores em infravermelho são comumente utilizadas, principalmente no âmbito comercial (ART; (FAT); LAB, ; SYSTEMS, 1996), devido à sua alta precisão. Partindo-se da noção de que ondas em infravermelho não são visíveis, e nem afetam a visão humana, os sensores infravermelhos são utilizados em conjunto com câmeras da seguinte forma: a luz do sensor é direcionada ao olho humano, e a reflexão é detectada pela câmera, facilitando a segmentação da pupila.

Certos problemas surgem da utilização destes métodos, principalmente devido à iluminação ambiente, que, se muito intensa, gera um efeito similar ao de um flash de câmera. Além disso, clarões podem ser gerados devido à incidência direta de luz, gerando falsos positivos.

## 3.3 Estimativa do Ponto Focal

A identificação da direção do olhar possui diversas aplicações, como no estudo da atenção humana (MURPHY-CHUTORIAN; DOSHI; TRIVEDI, 2007), auxílio de

portadores de deficiências locomotoras (ART; (FAT); LAB, ), interação humano-máquina (HYRSKYKARI, 2006), e pesquisas mercadológicas (CIAN; KRISHNA; ELDER, 2014).

O sentido da visão é gerado pela recepção de luz: fótons são refletidos no ambiente atravessam a córnea, são filtrados pela pupila, e são refratados pelo cristalino, chegando nas células fotorreceptoras localizadas na retina, posicionada na parte posterior do globo ocular. O olho humano pode ser tratado como um esferoide, ou, de forma simplificada, como uma esfera. Deste modo, adotando-se a representação esférica de representação, para estimar a direção do olhar deve-se calcular a direção do vetor tridimensional entre o centro da pupila e a retina. A estimativa de ponto focal é dada pela interseção dos vetores gerados pelos ângulos do olhar de ambos os olhos.

Apesar das simplificações geométricas aplicadas, o algoritmo é preciso o suficiente para gerar estimativas de baixo erro. Este algoritmo considera, no entanto, que a pessoa em estudo não possui estrabismo ou disfunção ocular similar. Nestes casos, cada olho possui uma angulação distinta, que pode ou não convergir para um ponto de interesse, e, conseqüentemente, o ponto focal não pode ser determinado corretamente sem a aplicação de heurísticas.





## 4 O Sistema

Visando gerar alta precisão, o sistema é dividido em três partes: (1) detecção da direção do olhar, na qual características e orientação facial são detectadas buscando a estimativa do ponto focal; (2) detecção de piscada; e (3) ajuste fino, no qual o erro da estimativa da etapa anterior é reduzida. A Figura 1 apresenta uma visão global da aplicação proposta, e mais detalhes serão dados nas seções a seguir. A implementação foi realizada em C++ com auxílio da biblioteca de visão computacional OpenCV (BRADSKI, 2000).

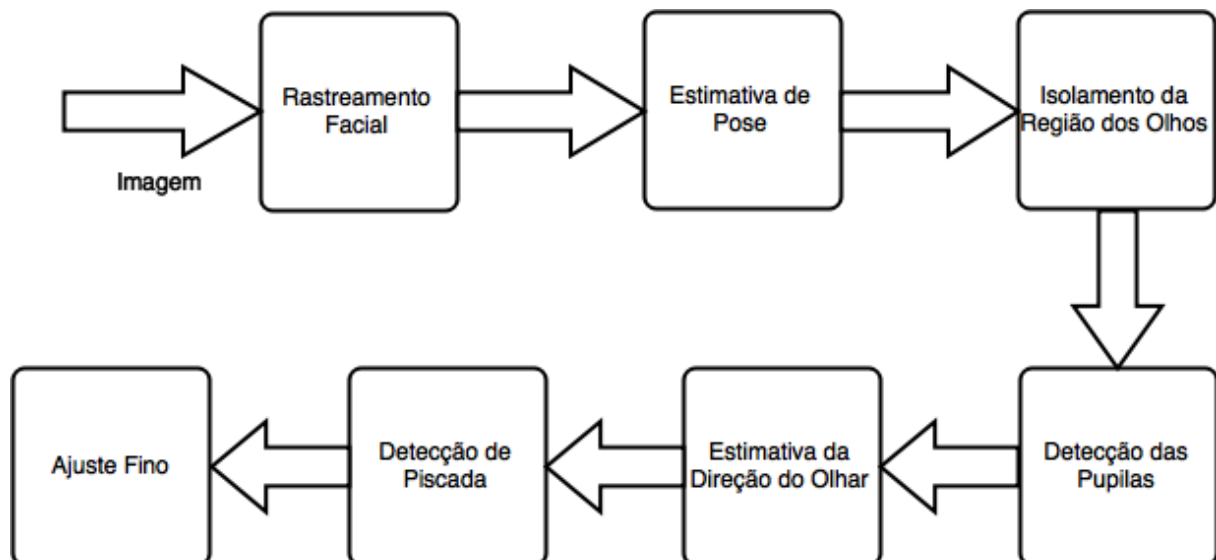


Figura 1 – Visão geral do sistema. A cada iteração do sistema, uma imagem do usuário é recebida e processada, gerando a estimativa de posição focal.

### 4.1 Modos de Operação

O sistema é controlado a partir de diferentes modos de operação que funcionam isoladamente e de formas distintas. Os modos são trocados através da aplicação do comando de troca de modo, isto é, uma piscada longa com ambos os olhos, ou através da finalização de execução do modo no qual o sistema se encontra através da execução de um clique. A Figura 2 apresenta a máquina de estados dos modos de operação. É válido ressaltar que não é possível inferir que os usuários sejam capazes de piscar ambos os olhos independentemente, então o comando de clique é realizado através do fechamento de qualquer um dos olhos.

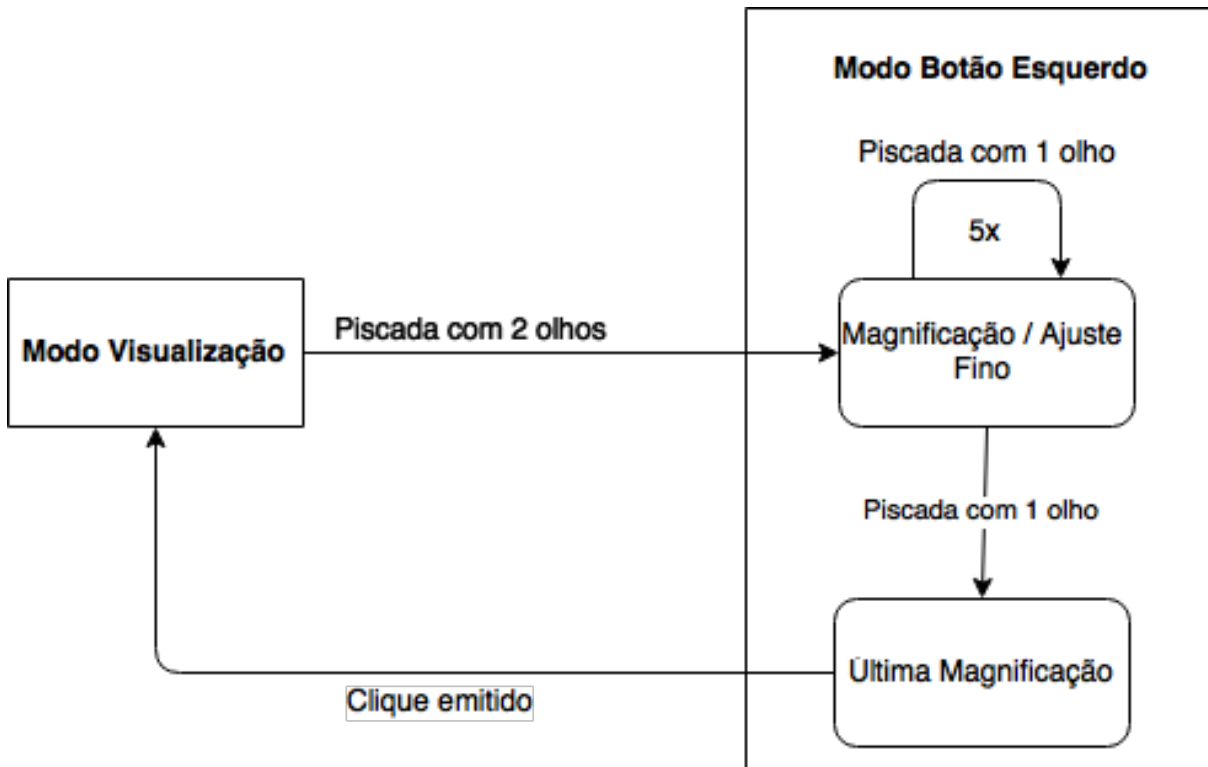


Figura 2 – Fluxograma dos modos de operação. A partir de um comando de mudança de modo de operação (piscada com 2 olhos), o sistema entra no modo Botão Esquerdo, no qual realizará magnificações na imagem até obter precisão suficiente para emitir um clique e retornar ao modo Visualização.

#### 4.1.1 Visualização

Como os olhos nunca se "desligam", o problema do Toque de Midas (JACOB, 1990) é gerado, isto é, o sistema não pode ficar em modo de operação a todo momento, já que o cursor do mouse se direcionaria para o ponto focal, atrapalhando a visualização da tela. Assim é necessária a existência deste modo, no qual o usuário tem interesse apenas no conteúdo atual da tela e, por isso, a interação com o mouse é desabilitada, e o cursor é escondido para não atrapalhar a leitura. Este é o modo no qual o sistema se encontra na maior parte do tempo.

#### 4.1.2 Botão Esquerdo

A utilização dos algoritmos de estimativa do ponto focal e de ajuste fino é efetuada ao entrar neste modo. O mouse é habilitado, e, após algumas iterações, o clique é efetuado, e o sistema retorna para o modo de visualização.

## 4.2 Cálculo da Direção do Olhar

O cálculo da direção do olhar é dividido nas seguintes etapas: rastreamento facial, estimativa de pose da cabeça, detecção das pupilas, estimativa da distância da tela e cálculo do ponto focal.

### 4.2.1 Rastreamento Facial e Estimativa de Pose

Para a estimativa de pose, optou-se pela utilização da biblioteca FaceTracker, a qual rastreia e detecta a posição e orientação da face através da técnica Arranjo Convexo Quadrático aplicada no domínio espaço-temporal (LUCEY *et al.*, 2010). Foi necessário, no entanto, adaptá-la para realizar uma nova detecção a cada 10 segundos, já que o rastreador frequentemente perde a referência da cabeça. Outros valores de intervalo de re-deteção são válidos, porém é importante notar que a re-deteção deve ser realizada numa frequência moderada, já que, devido ao custo computacional elevado da detecção facial, frequências altas diminuem a velocidade do sistema, e frequências baixas impedem o funcionamento pleno do sistema em caso de falha na detecção da face do usuário.

Apesar do FaceTracker possuir boa precisão no rastreamento geral das características faciais, a detecção da região ocular é defeituosa e os pontos detectados possuem alta imprecisão. Para amenizar tal problema, utilizou-se a biblioteca Flandmarks, uma implementação do algoritmo de detecção de pontos faciais descrito por Uricar *et al.* (UŘIČÁŘ; FRANC; HLAVÁČ, 2012) para detectar os cantos dos olhos de forma mais precisa.

O Flandmarks utiliza um classificador estruturado de saída baseado em Modelos de Peças Deformáveis (FELZENSZWALB *et al.*, 2010) para gerar um grafo contendo os cantos dos olhos, ponta do nariz, centro da face e cantos da boca. Da imagem da face são extraídas as características de Padrões Binários Locais e o vetor de deslocamento para determinar os possíveis pontos fiduciais. Então são aplicadas as restrições referentes ao formato do grafo através de programação dinâmica para detectar as posições mais prováveis dos pontos de interesse.

As características faciais extraídas pelos detectores descritos acima são mostradas na Figura 3: os círculos preenchidos são as características detectadas pelo Flandmark, os círculos vermelhos são gerados pelo FaceTracker, e os círculos brancos são as localizações estimadas das pupilas. Detalhes sobre a detecção das pupilas são fornecidos na próxima subseção.

### 4.2.2 Detecção de Pupilas

Primeiramente foi utilizado o método de localização do centro do olho através de histograma de gradientes orientados descrito por Timm *et al.* (TIMM; BARTH, 2011).

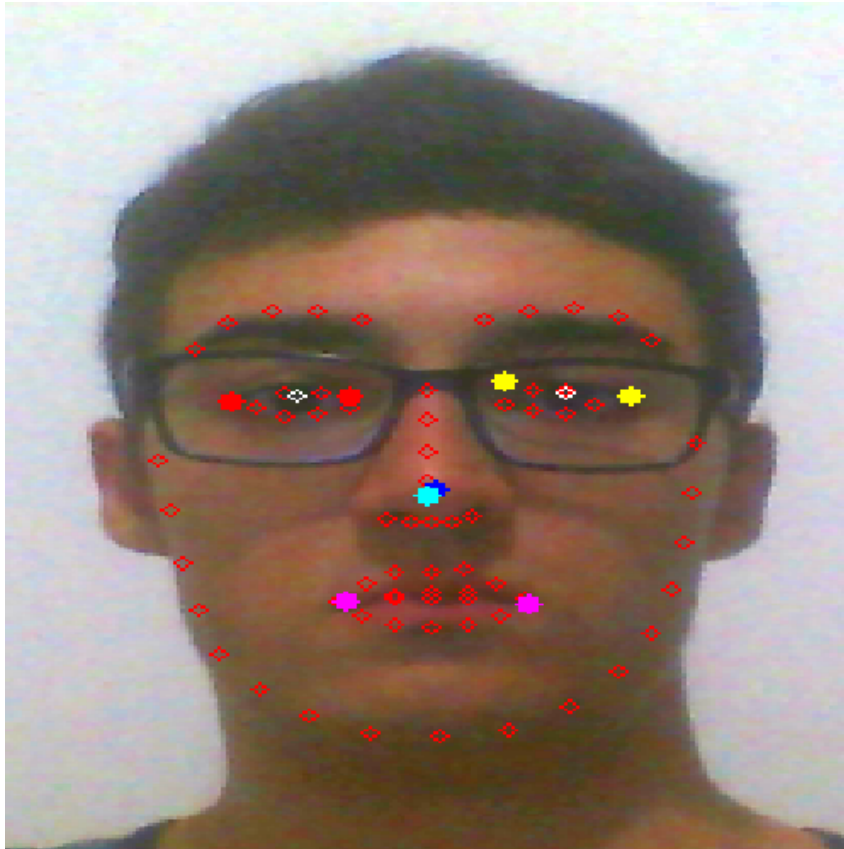


Figura 3 – Pontos detectados pelos diferentes detectores combinados.



(a) Imagem antes e depois do pré-processamento.

Figura 4 – Pontos detectados pelos diferentes detectores combinados.

Verificou-se, porém, que a detecção da pupila sofria muito com a ausência de iluminação na região. Para contornar tal problema, tratou-se o quadro a ser analisado com uma normalização prévia, de forma a ampliar o contraste entre os elementos. Desta forma pode-se diminuir a influência de gradientes menos significativos, e aumentar o peso de gradientes em regiões contrastantes, como pode ser observado na Figura 4.

Apesar da melhoria, o método ainda não apresentava resultados satisfatórios. Assim, aplicou-se um segundo algoritmo, o qual parte da certeza de que a pupila é um círculo mais escuro do que a região que a circunda. Uma visão geral do algoritmo é dada no

Algoritmo 1.

```

Entrada: imagem
início
  Gaussian Blur(imagem)
  Normalizar(imagem)
  escuro ← Encontrar o ponto mais escuro(imagem)
  Limiarização(imagem, escuro + kLimiar)
  contornos ← Encontrar contornos(imagem)
  circulo ← Encontrar circulo de maior raio(contornos)
  área ← Calcular área preenchida do circulo de maior raio(circulo)
  if área > área do circulo * kArea then
    | pupila ← centro do circulo
    | piscou ← FALSO
  end
  else
    | piscou ← VERDADEIRO
  end
fim

```

**Algoritmo 1:** Algoritmo para Detecção de Pupila e Piscada

Primeiramente, aplica-se um pré-processamento (Figura 5a) que diminui os ruídos na imagem e evita o excesso ou falta de claridade através, respectivamente, da aplicação de um filtro Gaussiano e normalização através da diminuição da quantidade de cores da imagem. O Filtro Gaussiano também é responsável por amenizar o brilho intenso causado pela reflexão do sol ou da luz do monitor em óculos através da diminuição do contraste da região.

Em seguida, partindo-se do pressuposto de que a imagem analisada contém apenas a região ocular e pequenas ocorrências de pele, encontra-se o ponto de valor mais escuro da imagem e aplica-se um limiar baseado em cores próximas de tal valor (*kLimiar*) para isolar os pontos mais escuros (Figura 5b). O valor utilizado no sistema para *kLimiar* foi 10.

Calcula-se então os contornos das regiões através do algoritmo descrito por Suzuki *et al.* (SUZUKI *et al.*, 1985), encontram-se os círculos que englobam tais contornos, e determina-se o círculo de maior raio. Compara-se então a área deste círculo com a área do contorno ao qual ele corresponde. Se a área do contorno é superior a um determinado percentual da área do círculo (*kArea*), uma pupila tem grande probabilidade de estar localizada dentro do círculo. Caso contrário, o olho está presumivelmente fechado (Figura 5c), e uma piscada foi realizada.

Utilizando-se o método descrito acima houve melhorias na detecção. Entretanto, a detecção de piscadas não se mostrou muito eficaz, pois sofre interferência pela existência

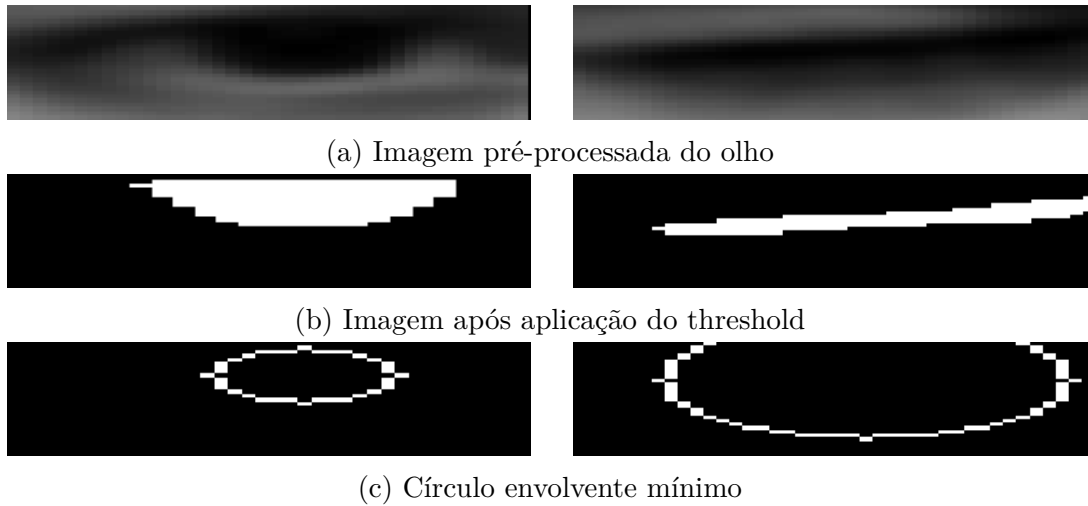


Figura 5 – Detecção de pupila e piscada.

de olheiras e baixo contraste entre esclera e pupila, detectando uma alta quantidade de falsos positivos em usuários que possuam estas características.

### 4.2.3 Estimativa da Distância da Tela

O cálculo da distância do usuário à tela  $d_1$  é realizado por uma estimativa a partir da relação entre o comprimento do olho do usuário  $s_1$  e o comprimento médio do olho humano  $s_2$  ( $24mm$ ) a uma distância conhecida da tela  $d_2$ , de acordo com a Equação 4.1:

$$d_1 = s_2 * d_2 / s_1 \quad (4.1)$$

Esta abordagem, no entanto, não é satisfatória devido à irregularidade da medição, e são estudadas alternativas para substituí-la.

### 4.2.4 Estimativa do Ponto Focal

Após a detecção da posição da pupila e estimativa da posição e orientação tridimensional da cabeça, a estimativa do ponto focal ocorre a partir de uma simplificação da teoria descrita no capítulo Aspectos Teóricos.

Inicialmente calcula-se a matriz de rotação da cabeça  $M_r$  através da estimativa de pose realizada na seção 4.2.1, de acordo com as Equações 4.2, 4.3, 4.4 e 4.5.:

$$rot_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos pose_x & -\sin pose_x \\ 0 & \sin pose_x & \cos pose_x \end{pmatrix} \quad (4.2)$$

$$rot_y = \begin{pmatrix} \cos pose_y & 0 & \sin pose_y \\ 0 & 1 & 0 \\ -\sin pose_y & 0 & \cos pose_y \end{pmatrix} \quad (4.3)$$

$$rot_z = \begin{pmatrix} \cos pose_z & -\sin pose_z & 0 \\ \sin pose_z & \cos pose_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4.4)$$

$$M_r = rot_z * rot_y * rot_x \quad (4.5)$$

Então calcula-se as posições tridimensionais das características faciais obtidas previamente (pupilas, cantos e centros dos olhos) baseadas na rotação da cabeça, assumindo-se que o olho humano é uma esfera perfeita de diâmetro aproximadamente igual à distância entre os cantos do olho. Para tal, para cada característica facial calcula-se a posição relativa dos pontos, rotaciona-se o ponto baseado na matriz de rotação, e calcula-se a posição absoluta dos pontos através da Equação 4.6. O ponto base é o centro do espaço cartesiano tridimensional no qual aplicar-se-á a rotação, *i.e.*, a característica facial mais próxima da câmera, já que a mesma tem maior chance de ter sido detectada corretamente.

$$pos3d = (pos2d - pontoBase) * M_r + pontoBase \quad (4.6)$$

Em seguida os ângulos entre o centro do olho e a pupila são calculados utilizando as Equações 4.7, 4.8 e 4.9.

$$distCentro = pupila3d - centroOlho3d \quad (4.7)$$

$$anguloHorizontal = \arctan \frac{distCentro_x}{centroOlho3d_z} \quad (4.8)$$

$$anguloVertical = \arctan \frac{distCentro_y}{centroOlho3d_z} \quad (4.9)$$

Por fim, sabendo-se a posição relativa ao usuário do monitor no eixo Z, pode-se então estimar as posições X e Y que interceptam o plano do monitor, através das Equações 4.10 e 4.11.

$$pontoEstimado_x = centroOlho3d_x - \tan anguloHorizontal * (posiçãoCabeça_z + centroOlho3d_z) \quad (4.10)$$

$$pontoEstimado_y = centroOlho3d_y - \tan anguloVertical * (posiçãoCabeça_z + centroOlho3d_z) \quad (4.11)$$

A Figura 6 apresenta imagens nas quais a direção do olhar foi estimada corretamente, e a Figura 7 apresenta imagens com estimativa incorreta. Nota-se que para corretude do cálculo, todas as coordenadas devem estar no mesmo sistema de medida. Para facilitar e simplificar os cálculos, tanto os pontos faciais, que estão em resolução de vídeo, quanto a distância da tela, que é medida em centímetros, foram mapeados para coordenadas em resolução do monitor.



Figura 6 – Estimativa correta de direção do olhar.

### 4.3 Detecção de Piscada

Conforme mencionado previamente, o algoritmo de detecção de pupila também pode ser utilizado para detecção de fechamento dos olhos, mas ele gera muitos falsos positivos, devido à baixa iluminação, baixo contraste entre esclera e pupila, ou quando as sobrancelhas estão localizadas dentro da região ocular estimada.



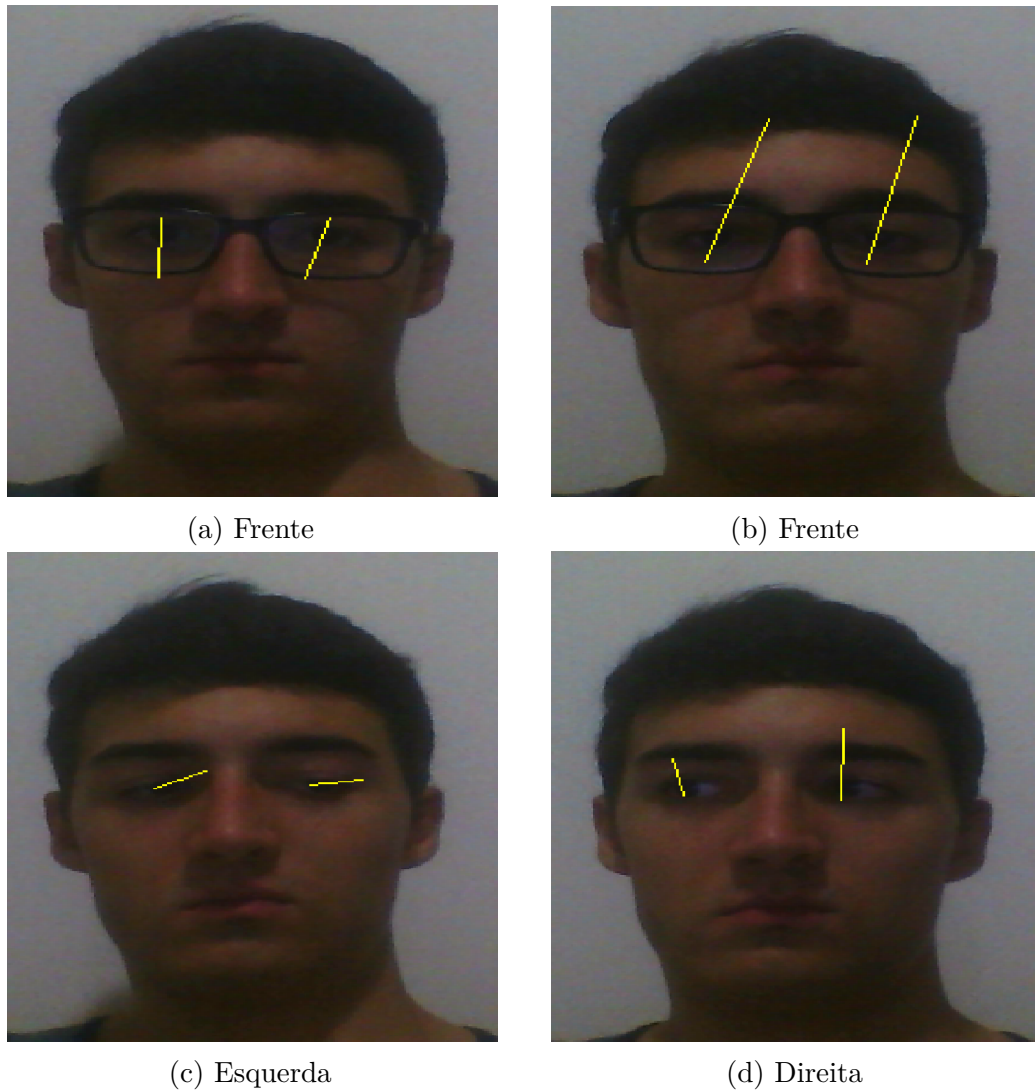


Figura 7 – Estimativa incorreta de direção do olhar.

Para diminuir a quantidade de falsos positivos, um classificador Haar treinado para olhos<sup>1</sup> foi aplicado para a detecção ocular. Caso ambos os detectores falhem na detecção do olho, é caracterizado o fechamento do respectivo olho. Isto permitiu a melhoria da qualidade da detecção de piscada, e, conseqüentemente, melhor utilização do sistema. Além disso, devido ao fato de que o ser humano pisca constantemente para lubrificar os olhos, um buffer acumulador é utilizado para detectar estado de fechamento dos olhos, de tal forma que um comando só seja confirmado se a piscada for mais longa que uma piscada de lubrificação.

<sup>1</sup> Um detector frontal de olhos com suporte para óculos baseado em árvore criado por Shameem Hameed (<http://umich.edu/~shameem>).

## 4.4 Ajuste Fino

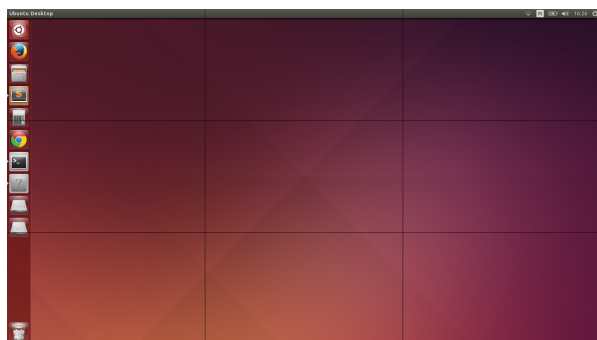
A estimativa do ponto focal não é exata devido à falta de precisão dos algoritmos de estimativa da pose, cálculo da distância da tela e detecção de pupila, e pode diferir significativamente da posição observada. Para amenizar o efeito do erro da estimativa, aplicou-se um algoritmo para ampliação de região de interesse. A abordagem consiste na visualização da tela como uma grade de magnificação que divide a tela em regiões menores, as áreas de zoom. Cada área de zoom, quando ampliada, torna-se a nova tela de visualização, a qual é dividida novamente até alcançar precisão suficiente (*ex*: a largura e a altura da área é menor do que um determinado limiar), momento no qual uma ação de clique é efetuada. A Figura 8 apresenta um exemplo de sequência de ampliações realizadas durante a execução de um clique.

Para uma velocidade de convergência máxima, uma grade de alta dimensionalidade deve ser usada, já que a quantidade de linhas e colunas da mesma é inversamente proporcional à quantidade de ampliações necessárias. Desta forma, assumindo-se uma precisão mínima  $p$  de 3 pixels, calcula-se a quantidade de ampliações  $A$  necessárias para alcance de tal precisão, satisfazendo a Equação 4.12 (supondo-se uma grade com largura e altura iguais, de valor  $d$ ):

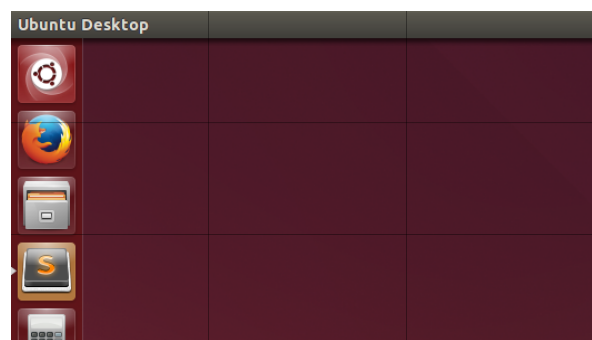
$$\log_d(\max(\text{larguraTela}, \text{alturaTela})/p) \leq A \quad (4.12)$$

Por exemplo, supondo-se a utilização de um monitor com a resolução de maior utilização atualmente, 1366x768 (vide Figura 9), e a utilização de uma grade 2x2, cada retângulo da grade possui metade da altura e largura da região de foco. Assim, para alcançar a precisão mínima são necessárias 9 ampliações. Para grades de outros tamanhos, como 3x3, 4x4, 5x5, 8x8, 21x21, 455x455, a quantidade de ampliações necessárias é, respectivamente, 6, 5, 4, 3, 2, 1 ampliações. Para este projeto, devido à alta discrepância entre o ponto focal correto e o estimado, utilizou-se uma grade de tamanho 3x3, como mostrado na Figura 8.

Algumas vezes, quando o usuário está longe da câmera, a resolução vertical do olho é muito pequena, não sendo suficiente para diferenciar muitas posições verticais, o que faz com que o usuário tenha que mover a cabeça na direção desejada. Para que pessoas com movimento de cabeça baixo ou nulo usem o sistema, a estimativa vertical da direção do olhar não é utilizada, o que faz com que o cursor mova-se lateralmente apenas. Para trocar de linha na grade, um limiar é aplicado a ambas as arestas verticais do monitor de tal forma que, se a posição do ponto focal está muito próxima da borda esquerda ou direita da tela, a linha em análise é modificada, respectivamente, para a anterior ou próxima.



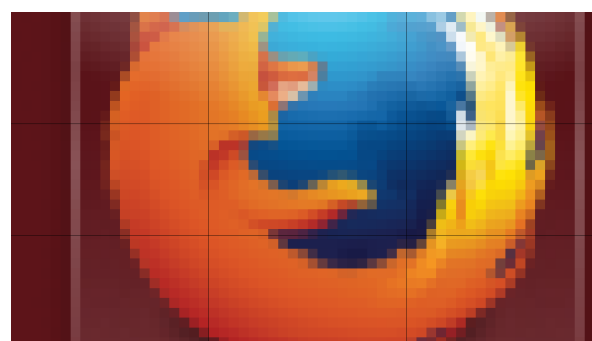
(a) Tela inicial.



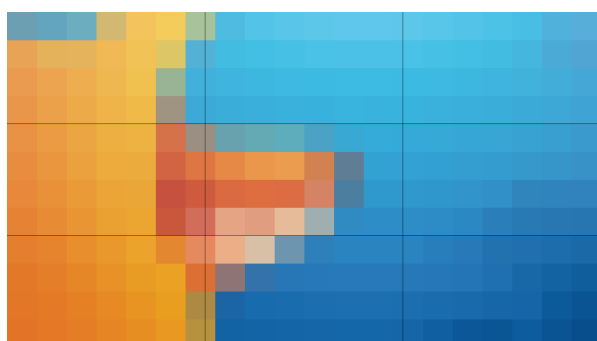
(b) Primeira ampliação.



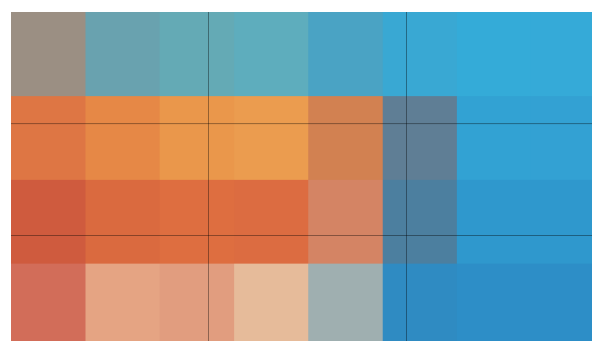
(c) Segunda ampliação.



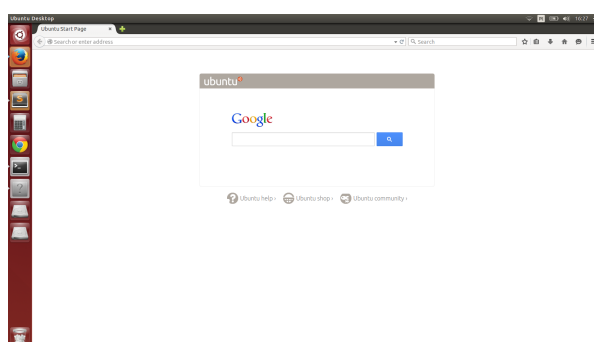
(d) Terceira ampliação.



(e) Quarta ampliação.



(f) Quinta ampliação.



(g) Sexta ampliação, que resulta em clique.

Figura 8 – Sequência de ampliações para a realização de um clique.

Most Used Screen Resolutions (April 19th 2015)

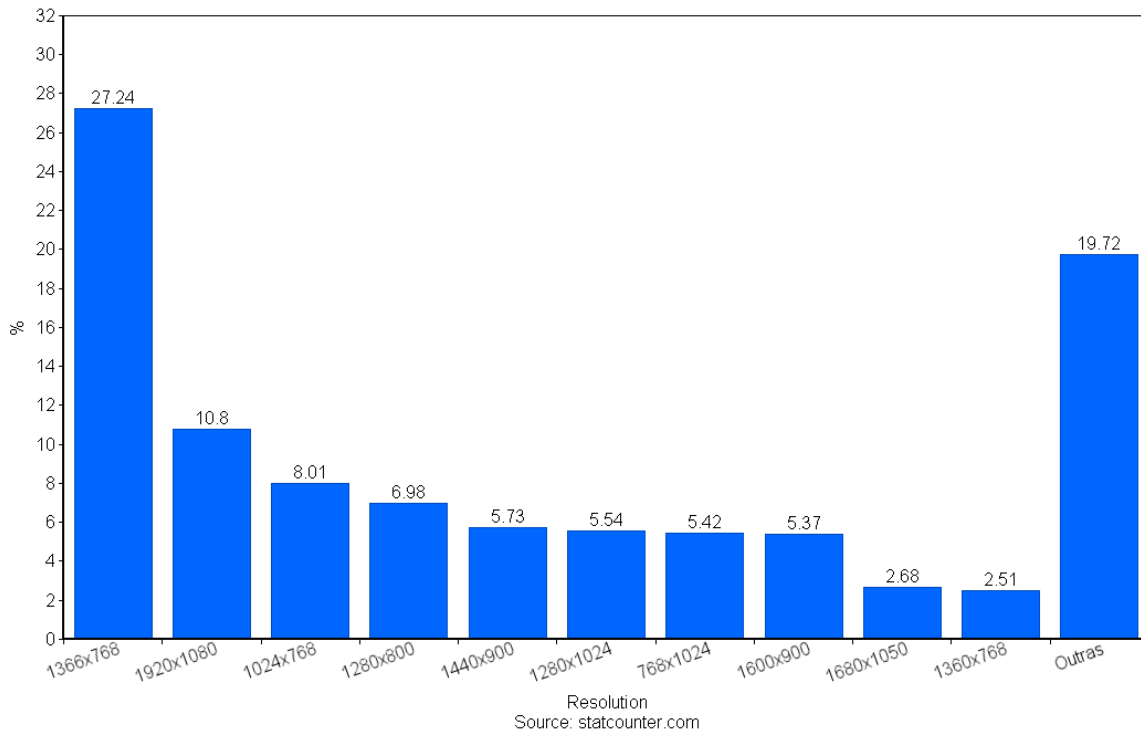


Figura 9 – Distribuição Atual de Resoluções de Monitores.

## 4.5 Filtragem

Devido à baixa qualidade da imagem e existência de ruídos na mesma, tanto a detecção de pontos faciais quanto a detecção da pupila tendem a oscilar. Desta forma, para melhorar a estimativa dos pontos de interesse, os dados são filtrados por um buffer de média: uma FIFO de tamanho limitado que armazena os valores de posição detectados e estima a provável posição desejada através da média de tais valores.

O tamanho do buffer não deve ser muito pequeno nem muito grande, pois pode, respectivamente, não reduzir significativamente a interferência dos ruídos ou tornar o sistema lento e pouco responsivo. No sistema foram utilizados filtros de 50 elementos, e a filtragem permitiu uma melhoria significativa, reduzindo a interferência de erros na localização das características faciais, como pode ser visto na Figura 10, que apresenta os pontos estimados em um experimento no qual o usuário olhou para um ponto fixo na tela. A linha contínua representa o ponto estimado, e a linha tracejada representa o ponto após a aplicação do filtro. É válido notar a diferença de precisão alcançada nos dois eixos: enquanto no eixo horizontal a discrepância do ponto estimado para o ponto real é inferior a 10%, o erro no eixo vertical supera 20%.

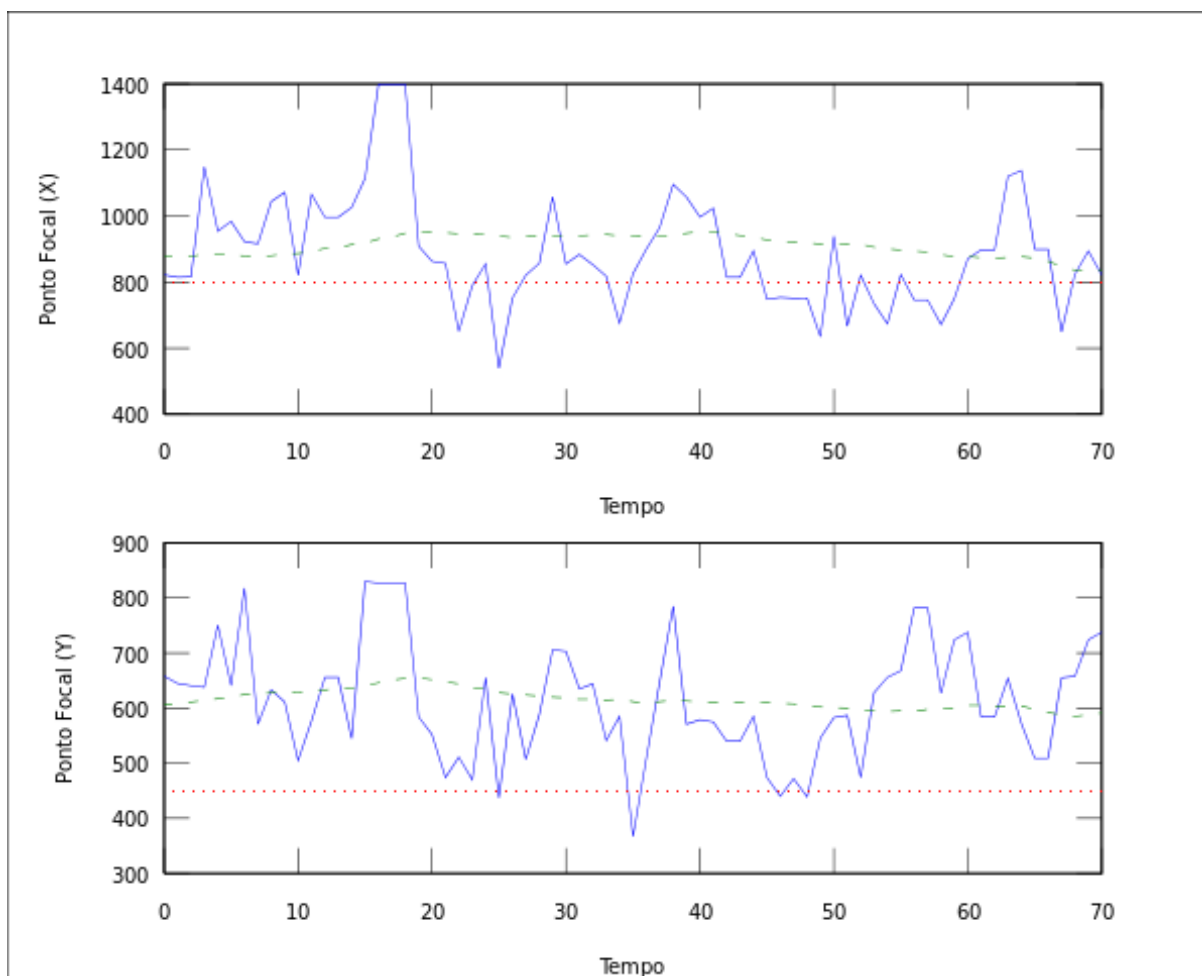


Figura 10 – Comparação entre as estimativas de pontos focais com e sem filtragem. A linha tracejada representa o ponto estimado após a aplicação do filtro. A reta pontilhada representa a posição real do ponto.



## 5 Resultados

O sistema foi testado em um computador com Ubuntu 14.04 com processador Intel Core i7-3615QM de 2.3GHz e processador gráfico NVIDIA GeForce GT 630 M. A resolução do monitor utilizada foi 1600x900, e a resolução da câmera foi 640x480. Como a detecção do FaceTracker é mais computacionalmente custosa que o rastreamento, há uma queda de desempenho quando nenhum usuário é detectado. Durante utilização normal, o sistema obteve desempenho razoável, alcançando uma média de 20 quadros por segundo.

Visando a validação da técnica, um sistema online foi desenvolvido para estimar a precisão do clique. Esse consiste em mostrar vinte alvos de 50x50 pixels em lugares randômicos da tela, e o usuário é requisitado a clicar o mais próximo do centro que ele puder. A Figura 11 mostra o sistema online em execução.

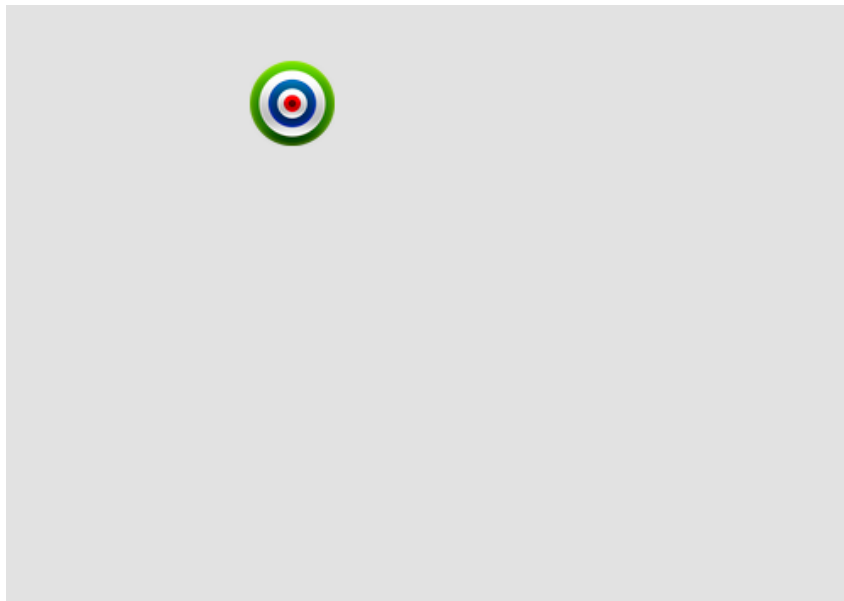


Figura 11 – Sistema online: O usuário clica o mais próximo do centro do alvo para gerar a estimativa de precisão.

Para avaliação da precisão média na utilização de um mouse/touchpad foi realizado um experimento com 20 pessoas de diferentes idades. Assumindo-se que a mediana dos erros dos resultados representa a precisão média dos usuários, verificou-se que a precisão média para este experimento foi de 3 pixels. O mesmo experimento foi então aplicado a um único usuário usando o sistema proposto, produzindo uma precisão de 2 pixels. Estes resultados, mostrados na Figura 12a, indicam que o sistema é aplicável, e que pode ser usado como um substituto dos métodos convencionais para ajudar pessoas com deficiências. Devido à natureza quantitativa dos experimentos, é necessária a realização de mais testes para validar a utilização do sistema por outros usuários.

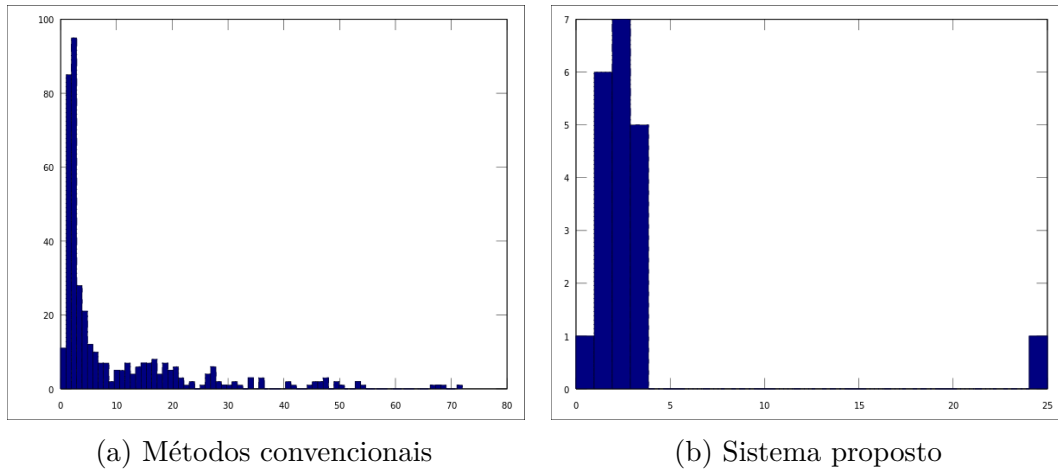


Figura 12 – Histogramas das precisões alcançadas nos experimentos.

A estimativa de pose gerada pelo FaceTracker proporcionou resultados robustos, com erro médio estimado em menos de cinco graus. Contudo, o rastreador perde a posição correta da face às vezes, então a detecção facial é forçada a cada 10 segundos para contornar este problema, ao custo de perda de velocidade de execução.

A detecção das regiões oculares e respectivos pontos de interesse pela Flandmark, após a realização de filtragem para remover inconsistência de dados, foi precisa, gerando erros significativos apenas em faces rotacionadas no eixo Z (profundidade), ou com rotação superior a 35 graus no eixo Y (vertical), devido à auto oclusão na região da face.

O algoritmo para rastreamento de pupila fornece alta precisão em ambientes pouco iluminados, mas pode gerar resultados errôneos quando a rotação da cabeça é superior a 40 graus, ou quando existem elementos oclusivos na imagem, como cabelo e óculos. Além disso, baixo contraste entre a esclera e a pupila impede a detecção.

A detecção do piscar dos olhos falha quando há auto sombreamento da região ocular, e a ocorrência de falsos positivos quebra o sistema, já que eles podem ativar comandos em momentos indesejáveis, transladando o cursor do mouse para posições indesejadas irrecuperavelmente. Alguns resultados da detecção de piscada são mostrados na Figura 13.

A implementação da correção de estimativa do sistema através de grades e ampliações permitiu que, apesar da estimativa da direção do olhar apresentar grandes erros devido ao acumulo de erros das etapas anteriores, fosse possível a utilização do sistema, e os pontos desejados podem ser obtidos com precisão próxima a 100%, como pode ser visto na Figura 12b.





Figura 13 – Detecção de fechamento de olhos. Um quadrado verde é mostrado no lado no qual a piscada é detectada.



## 6 Conclusão e Trabalhos Futuros

Neste trabalho foram aplicados conceitos de rastreamento de direção do olhar, e uma nova técnica para correção de estimativa foi introduzida, a qual reduz erros causados pela falta de precisão de detectores de características faciais ao custo de tempo de execução.

Apesar da falta de precisão da estimativa do ponto focal, o algoritmo adotado para correção forneceu uma precisão próxima a 100%. No entanto, devido à utilização de algoritmos de detecção facial e de pupilas susceptíveis a erros, o software frequentemente perde a localização facial. Além disso, como a estimativa de distância do usuário da tela é imprecisa, o erro na detecção do ponto focal torna-se alto, o que aumenta o tempo de convergência para a posição correta.

Para contorno de tais problemas, é necessária a utilização de algoritmos melhores para a detecção facial/ocular e orientação da cabeça, como o Método de Descida Supervisionada (XIONG; TORRE, 2013); e para a detecção de pupila, como o de segmentação de íris através do operador integrodiferencial de Daugman (DAUGMAN, 2004). Para melhorar o cálculo de distância da tela, uma possibilidade é a realização de calibração a partir da criação de um ponto de interesse na tela. A direção do olhar do usuário deve ser então calculada utilizando valores aleatórios de distância até o ponto focal estimado ser próximo do ponto de calibragem.

Outras medidas válidas para a melhoria do sistema são a aplicação de filtros de Kalman para análise e predição de movimentação ocular, aumentando a precisão da detecção (KOMOGORTSEV; KHAN, 2007; ABD-ALMAGEED; FADALI; BEBIS, 2002; GRINDINGER, 2006), e o incremento da velocidade de aquisição da imagem da área de trabalho para visualização em tempo real durante a movimentação do mouse.

Futuramente outros modos de operação podem ser adicionados, permitindo que o usuário efetivamente consiga utilizar o computador sem a necessidade de assistência humana. Tais modos devem simular possíveis comandos de mouse e teclado, como a troca entre janelas ativas, clique duplo, clique com o botão direito, e digitação. O último pode ser implementado através da utilização de um algoritmo de predição de palavras, como sugerido em Ward *et al.* (WARD; BLACKWELL; MACKAY, 2000).



# Referências

- ABD-ALMAGEED, W.; FADALI, M. S.; BEBIS, G. A non-intrusive kalman filter-based tracker for pursuit eye movement. In: IEEE. *American Control Conference, 2002. Proceedings of the 2002.* [S.l.], 2002. v. 2, p. 1443–1447. Citado na página 49.
- ARAR, N. M.; GAO, H.; THIRAN, J.-P. Robust gaze estimation based on adaptive fusion of multiple cameras. In: *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition.* [S.l.: s.n.], 2015. Citado na página 24.
- ART, F.; (FAT), O. T.; LAB the G. R. *Eye Writer*. Disponível em: <<http://eyewriter.org/>>. Citado 2 vezes nas páginas 28 e 29.
- BOULAY, G. Eye pose tracking & gaze estimation. *Institut National des Sciences Appliquées de Lyon, France. Département Télécommunications–Services & Usages*, 2008. Citado na página 23.
- BRADSKI, G. opencv. *Dr. Dobb’s Journal of Software Tools*, 2000. Citado na página 31.
- CHEN, J. et al. A robust 3d eye gaze tracking system using noise reduction. In: ACM. *Proceedings of the 2008 symposium on Eye tracking research & applications.* [S.l.], 2008. p. 189–196. Citado na página 24.
- CIAN, L.; KRISHNA, A.; ELDER, R. S. This logo moves me: Dynamic imagery from static images. *Journal of marketing research*, American Marketing Association, v. 51, n. 2, p. 184–197, 2014. Citado na página 29.
- COOTES, T. F.; EDWARDS, G. J.; TAYLOR, C. J. Active appearance models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, IEEE, n. 6, p. 681–685, 2001. Citado na página 26.
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: IEEE. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on.* [S.l.], 2005. v. 1, p. 886–893. Citado na página 26.
- DAUGMAN, J. How iris recognition works. *Circuits and Systems for Video Technology, IEEE Transactions on*, IEEE, v. 14, n. 1, p. 21–30, 2004. Citado na página 49.
- DEMENTHON, D. F.; DAVIS, L. S. Model-based object pose in 25 lines of code. *International journal of computer vision*, Springer, v. 15, n. 1-2, p. 123–141, 1995. Citado na página 26.
- DUDA, R. O.; HART, P. E. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, ACM, v. 15, n. 1, p. 11–15, 1972. Citado na página 27.
- FELZENSZWALB, P. F. et al. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 32, n. 9, p. 1627–1645, 2010. Citado na página 33.

FISCHLER, M. A.; BOLLES, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, ACM, v. 24, n. 6, p. 381–395, 1981. Citado na página 23.

FUJITSU. *Fujitsu's Eye Tracker*. 2012. Disponível em: <<http://www.fujitsu.com/global/about/resources/news/press-releases/2012/1002-02.html>>. Citado 2 vezes nas páginas 21 e 24.

GRINDINGER, T. *Eye movement analysis & prediction with the kalman filter*. Tese (Doutorado) — Clemson University, 2006. Citado na página 49.

HANSEN, D. W.; JI, Q. In the eye of the beholder: A survey of models for eyes and gaze. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 32, n. 3, p. 478–500, 2010. Citado na página 27.

HYRSKYKARI, A. *Eyes in attentive interfaces: Experiences from creating iDict, a gaze-aware reading aid*. [S.l.]: Tampereen yliopisto, 2006. Citado na página 29.

JACOB, R. J. What you look at is what you get: eye movement-based interaction techniques. In: ACM. *Proceedings of the SIGCHI conference on Human factors in computing systems*. [S.l.], 1990. p. 11–18. Citado na página 32.

KAWATO, S.; TETSUTANI, N. Detection and tracking of eyes for gaze-camera control. *Image and Vision Computing*, Elsevier, v. 22, n. 12, p. 1031–1038, 2004. Citado na página 24.

KO, J.-G.; YU, J.-H.; JEONG, K.-I. Facial feature detection and head orientation based gaze tracking. In: WORLD SCIENTIFIC AND ENGINEERING ACADEMY AND SOCIETY (WSEAS). *Proceedings of the 2005 WSEAS international conference on Dynamical systems and control*. [S.l.], 2005. p. 415–420. Citado na página 23.

KOMOGORTSEV, O. V.; KHAN, J. I. Kalman filtering in the design of eye-gaze-guided computer interfaces. In: *Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments*. [S.l.]: Springer, 2007. p. 679–689. Citado na página 49.

LI, D.; BABCOCK, J.; PARKHURST, D. J. openeyes: a low-cost head-mounted eye-tracking solution. In: ACM. *Proceedings of the 2006 symposium on Eye tracking research & applications*. [S.l.], 2006. p. 95–100. Citado na página 23.

LUCEY, S. et al. Non-rigid face tracking with enforced convexity and local appearance consistency constraint. *Image and vision computing*, Elsevier, v. 28, n. 5, p. 781–789, 2010. Citado na página 33.

MAJARANTA, P.; BULLING, A. Eye tracking and eye-based human–computer interaction. In: *Advances in Physiological Computing*. [S.l.]: Springer, 2014. p. 39–65. Citado na página 23.

MARTINS, P. A. D. *Active appearance models for facial expression recognition and monocular head pose estimation*. 2008. Citado na página 26.

MURPHY-CHUTORIAN, E.; DOSHI, A.; TRIVEDI, M. M. Head pose estimation for driver assistance systems: A robust algorithm and experimental evaluation. In: IEEE. *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. [S.l.], 2007. p. 709–714. Citado na página 28.

- MURPHY-CHUTORIAN, E.; TRIVEDI, M. M. Head pose estimation in computer vision: A survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 31, n. 4, p. 607–626, 2009. Citado na página 25.
- SUZUKI, S. et al. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, Elsevier, v. 30, n. 1, p. 32–46, 1985. Citado na página 35.
- SYSTEMS, E. D. *Eye Tracking Technology for OEMs*. 1996. Disponível em: <<http://www.eyetechds.com/>>. Citado na página 28.
- TIMM, F.; BARTH, E. Accurate eye centre localisation by means of gradients. In: *VISAPP*. [S.l.: s.n.], 2011. p. 125–130. Citado 2 vezes nas páginas 27 e 33.
- TOBII. *Tobii*. 2010. Disponível em: <[www.tobii.com](http://www.tobii.com)>. Citado 2 vezes nas páginas 21 e 24.
- TRAN, D. T. et al. Robust methods for head orientation estimation and user identification based on hog and codebook. In: IEEE. *System Integration (SII), 2011 IEEE/SICE International Symposium on*. [S.l.], 2011. p. 224–229. Citado na página 26.
- UŘIČÁŘ, M.; FRANC, V.; HLAVÁČ, V. Detector of facial landmarks learned by the structured output svm. *VISAPP*, v. 12, p. 547–556, 2012. Citado na página 33.
- VALENTI, R.; SEBE, N.; GEVERS, T. Combining head pose and eye location information for gaze estimation. *Image Processing, IEEE Transactions on*, IEEE, v. 21, n. 2, p. 802–815, 2012. Citado na página 23.
- VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: IEEE. *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. [S.l.], 2001. v. 1, p. I–511. Citado na página 25.
- WARD, D. J.; BLACKWELL, A. F.; MACKAY, D. J. Dasher—a data entry interface using continuous gestures and language models. In: ACM. *Proceedings of the 13th annual ACM symposium on User interface software and technology*. [S.l.], 2000. p. 129–137. Citado na página 49.
- XIONG, X.; TORRE, F. De la. Supervised descent method and its applications to face alignment. In: IEEE. *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. [S.l.], 2013. p. 532–539. Citado na página 49.
- YUILLE, A. L.; HALLINAN, P. W.; COHEN, D. S. Feature extraction from faces using deformable templates. *International journal of computer vision*, Springer, v. 8, n. 2, p. 99–111, 1992. Citado na página 27.
- ZIELINSKI, P. *Opengazer: open-source gaze tracker for ordinary webcams*. Disponível em: <<http://www.inference.phy.cam.ac.uk/opengazer/>>. Citado na página 21.