
**Universidade Católica Dom Bosco**
Centro de Ciências Exatas e Tecnológicas
Curso de Engenharia de Computação

# Estudo de Técnicas de Rastreamento das Mãos para o Desenvolvimento de Interfaces Homem-Máquina

Álvaro Roberto Silvestre Fialho

Prof. Orientador: Dr. Hemerson Pistori

*Relatório Final submetido como um dos requisitos para a obtenção do grau de Engenheiro de Computação.*

UCDB - Campo Grande - MS - Novembro/2004

# Agradecimentos

Agradeço primeiramente a Deus, por ter me dado vida, saúde e sabedoria no decorrer deste trabalho.

A meus pais Alvaro e Cristina, e à minha namorada Camila, por terem me provido estabilidade emocional e carinho nas horas de correria e desespero, e principalmente nas horas em que nada funcionava, meu Athlon só esquentava, etc. Ao meu irmão Alesi, por ter compreendido realmente que eu precisava do computador, e ter ficado sem jogar Counter-Strike enquanto eu estava em casa, para deixar o computador disponível para mim.

Agradeço também ao meu orientador, professor Hemerson Pistori, pelas horas de lições, tanto teóricas como também de moral e, principalmente, de motivação, se tornando um grande amigo. A todos os integrantes do grupo SIGUS, que também me ajudaram a solucionar vários problemas durante a etapa final do projeto.

Sou muito grato também a todos os meus colegas de turma e amigos da Engenharia: sem estes companheiros de madrugadas passadas em claro estudando (ou se divertindo), as coisas com certeza teriam sido muito mais difíceis.

# Resumo

Os principais meios de entrada de dados com os quais o homem interage com o computador são: teclado, *mouse* e *joystick*. Reconhecimento de gestos das mãos, expressão facial (emoção), direção do olhar, entre outros, são campos que vêm sendo estudados com o objetivo de desenvolver novas maneiras de se interagir com o computador. Através do rastreamento do movimento das mãos é possível, por exemplo, controlar os canais e volume de uma televisão, ou mesmo fazer o reconhecimento de sinais, como os da Língua Brasileira de Sinais (LIBRAS). Este trabalho tem como objetivo estudar as principais técnicas de rastreamento das mãos baseadas apenas em visão computacional, visando implementar o rastreamento das mãos em um sistema que converte LIBRAS em texto.

# Abstract

The primary means of input in Human Computer Interaction (HCI) are keyboards, mouses and joysticks. Gesture recognition, emotion recognition, eyes-gaze direction, among others, are fields that are being studied with the objective of developing new ways to interact with the computer. By tracking hands movement it is possible, for example, to control a television tuner, or even to recognise language signals, such as the signals of the Brazilian Sign Language(LIBRAS). This work has the objective of studying the principal hand tracking techniques based only on computer vision, in order to develop the hand tracking in a system which converts LIBRAS into text.

# Conteúdo

# Lista de Figuras

# Capítulo 1

# Introdução

Os principais meios de entrada de dados com os quais o homem interage com o computador são: teclados, *mouse* e *joystick*. Entretanto, assim como a complexidade das aplicações vêm crescendo ao longo do tempo, cresce também a necessidade por dispositivos de interação diferentes dos convencionais. De acordo com Bebis et al, 2002 [5], uma boa opção para suprir essas necessidades é importar os meios naturais de comunicação e interação humana para o meio de interação homem-computador. Neste trabalho nos concentramos no rastreamento das mãos como dispositivo de interação homem-máquina, estudando as principais técnicas utilizadas na busca pelas mãos na imagem.

## 1.1 Motivação

Vários dispositivos de rastreamento das mãos já estão disponíveis no mercado, dentre os quais podemos citar os dispositivos magnéticos (ex.: Polhemus Fastrack [1]), os dispositivos de interação 3D (ex.: SpaceBall [2]), e os dispositivos com sensores eletro-mecânicos (ex.: CyberGlove [3]).

Apesar dessa variedade de novos meios, a interação Homem-Computador ainda difere muito da interação Homem-Homem. A interação natural entre os homens não envolve nenhum dispositivo porque nós temos a habilidade de *sentir* o ambiente com os olhos e os ouvidos. O ideal seria que os computadores pudessem imitar essas habilidades com câmeras e microfones, o que proporcionaria duas grandes vantagens. Primeiramente, eles poderiam ocupar bem menos espaço em uma mesa de trabalho, por exemplo, visto

[1] Informações sobre o Fastrack: `http://www.polhemus.com/fastrak.htm`

[2] Informações sobre o SpaceBall: `http://www.abs-tech.com/produtos/descontinuados/spaceball_4000_flx/spaceball_4000_flx.html`

[3] Informações sobre a CyberGlove: `http://www.simsol.co.uk/cyberglove.shtml`

que uma webcam e um microfone ocupam muito menos espaço do que um teclado e um mouse. A outra vantagem é que seria muito mais fácil utilizá-los, visto que os gestos, as expressões faciais, as palavras faladas, etc, serviriam como entrada de dados, tornando a interação homem-computador muito semelhante à comunicação natural entre homens.

Reconhecimento de gestos, de línguas de sinais, de emoções, rastreamento de mãos e braços ou mesmo do corpo como um todo, são exemplos de diversas áreas de pesquisa, nas quais são produzidas ferramentas que podem ser utilizadas na comunicação do homem com o computador. Além disso, com o avanço na velocidade de processamento e nas tecnologias de visualização, uma quantidade cada vez maior de métodos de interação sofisticados estão sendo criados, como por exemplo, o mapeamento de vídeo, a realidade virtual imersiva e a telepresença, que são coletivamente chamados de realidade virtual, e que requerem uma estimativa precisa da posição do corpo humano, de acordo com Bebis et al, 2002 [5]. Através do rastreamento do movimento das mãos é possível, por exemplo, controlar os canais e volume de uma televisão (Freeman e Weissman, 1995 [19]), ou mesmo reconhecer sinais, como os da Língua Brasileira de Sinais [4] (Pistori e Neto, 2004 [53]).

Existem muitos casos nos quais o uso das mãos como interface homem-máquina é muito mais prático do que o uso dos dispositivos convencionais, como por exemplo:

- Durante uma apresentação, o apresentador não precisa ir e voltar até o computador para passar para o próximo slide, podendo fazer apenas um sinal com a mão;
- Dispositivos móveis, com espaço muito limitado para interfaces, poderiam ser operados com gestos;
- Controles remotos, interruptores de luz, etc, poderiam ser trocados pelas mãos;
- Durante uma videoconferência, a câmera poderia ser direcionada a filmar uma determinada região para onde o usuário aponta (*Olhe para essa região.*);
- Robôs ou outras máquinas poderiam ser controladas através de gestos.

Uma aplicação interessante, proposta em Pistori et al, 2004 [55], é o reconhecimento de línguas de sinais. Em um futuro não muito longínquo um

[4]LIBRAS: Língua Brasileira de Sinais - oficialmente reconhecida como meio legal de comunicação pela lei federal número 10.436 de 24 de abril de 2002.

surdo poderia realizar os sinais em frente a um dispositivo, que reconheceria os sinais (visão computacional) e os transformaria em áudio (geração de fala), possibilitando assim o diálogo entre surdos e pessoas que não conhecem as línguas de sinais. Outra aplicação, citada em vários artigos, é a substituição do mouse físico. É possível movimentar o cursor, clicar, arrastar objetos, fazer todo o serviço do mouse, através do rastreamento das mãos e do reconhecimento de gestos, conforme demonstrado em Lin et al, 2002 [39] e em Ianizzoto et al, 2001 [28].

Com a utilização das mãos como interface, é possível criar computadores que não são *vistos* como tal. Sem monitor, mouse e teclado, um computador pode ser escondido em vários lugares, como eletrodomésticos, carros, máquinas em geral e brinquedos. As principais vantagens da utilização da visão computacional como interface, sobre os tradicionais botões e interruptores são as seguintes:

- Sistemas podem ser operados à distância;
- Sistemas podem ser protegidos do vandalismo, criando-se uma margem de segurança entre o usuário e o equipamento;
- Projetos de equipamentos muito diferentes dos atuais podem ser construídos, como por exemplo uma televisão sem nenhum botão;
- Em combinação com o reconhecimento da fala, a interação homem-máquina se torna muito mais simples.

Existe também uma variedade de aplicações que podem ser construídas em conjunto com um projetor. Objetos virtuais são projetados na parede, por exemplo, e podem ser diretamente manipulados com as mãos. A seguir citamos alguns exemplos:

- Várias pessoas podem trabalhar simultaneamente com objetos projetados na parede, como no *BrainStorm*, aplicação proposta por Hardenberg e Bérard, 2001 [24], na qual vários tópicos são projetadas na parede em uma reunião, os participantes se dirigem até a parede e, com as mãos, organizam e ordenam as palavras projetadas.
- Dispositivos físicos podem ser substituídos por dispositivos virtuais controlados pelas mãos. Por exemplo, em Lin et al, 2002 [39] foi desenvolvido um sistema virtual de DJ, que substitui a mesa de som (física).
- Se o projetor e a câmera são instalados em um local que o usuário não consegue acessar, uma interface praticamente indestrutível pode ser construída. O que o usuário pode tocar é apenas a parede onde a interface é projetada.


Figura 1.1: Aplicações utilizando a mão como interface homem-máquina. (a) Controlando o cursor do mouse com o dedo. (b) Pintando uma figura com o dedo. (c) Controlando uma apresentação com posturas da mão. (d) Várias mãos reorganizando ítens projetados. Imagens reproduzidas de Hardenberg e Bérard, 2001 [24].

Enquanto algumas dessas aplicações podem parecer muito futuristas, outras podem ser um tanto práticas. É importante notar que sistemas de reconhecimento de fala também proporcionam os benefícios listados acima. A vantagem das técnicas baseadas na visão é que não perturbam o fluxo de uma conversa (em uma apresentação, por exemplo) e funcionam muito bem em ambientes barulhentos (em um espaço público, por exemplo).

Algumas técnicas fazem uso de luvas coloridas para facilitar o rastreamento das mãos na imagem, como em Bebis et al, 2002 [5]. Outras utilizam hardware específico, podendo-se citar como exemplo o *FingerMouse*, um sistema com duas câmeras e um sensor de distância, sempre visando obter um melhor resultado, conforme Hamette et al, 2002 [22]. Segundo Lu et al, 2002 [40], existem vários dispositivos baseados em luvas com sensores eletromecânicos, que capturam os movimentos das mãos diretamente, no entanto esses sensores são caros e difíceis de serem utilizados. Além disso, Iannizzotto et al, 2001 [28] cita que as técnicas de rastreamento baseadas apenas na visão computacional estão sendo as mais utilizadas, já que não requerem nenhum dispositivo adicional (luvas, etc.), e podem ser implementadas com imagens capturadas de uma simples webcam, tornando o seu baixo custo outro grande atrativo. Com o constante barateamento dos dispositivos de captura de imagem, webcams já podem ser consideradas recurso básico dos computadores, assim como mouse, teclado, caixas de som, etc. Essa é uma das motivações para este trabalho, pois com a utilização de recursos acessíveis, é possível se obter resultados interessantes.

Existe um número muito grande de pessoas trabalhando com soluções de rastreamento baseadas em visão, com uma quantidade vasta de técnicas

diferentes já desenvolvidas, como por exemplo: soluções baseadas em filtros de Kalman (Vaillant e Darmos, 1995 [76]), detecção da cor da pele (Hongo et al, 2000 [26]), ou mesmo utilizando modelos de mãos 3D (Heap e Hogg, 1996 [25]). Esse é outro motivo pelo qual esta pesquisa está sendo desenvolvida, visando se tornar uma fonte para todos aqueles que se interessam por esse campo de atuação da Visão Computacional.

Foi desenvolvido por Pistori e Neto, 2003 [54] um sistema que atua como um conversor LIBRAS-texto, em que o usuário realiza o gesto de uma das letras do alfabeto LIBRAS, a câmera captura o gesto e o transforma em texto no computador. Este protótipo já funciona muito bem reconhecendo os padrões (gestos), entretanto é necessário que a mão esteja em uma posição pré-fixada, e o fundo necessita ser uniforme. Outra forte motivação para este trabalho é que, após ter estudado diversas técnicas, o conhecimento adquirido foi suficiente para adicionar à este conversor este novo recurso: o rastreamento das mãos, ampliando assim a sua aplicabilidade, e facilitando o seu uso.

## 1.2 Objetivos

Este trabalho têm, como um dos objetivos, a realização de um estudo abrangente das técnicas de rastreamento das mãos, servindo, no futuro, como uma fonte para todos aqueles que se interessam por esse campo de atuação da Visão Computacional. O outro objetivo é adicionar ao conversor LIBRAS-texto o rastreamento das mãos, aumentando a sua aplicabilidade.

## 1.3 Metodologia

A posição das mãos na imagem pode ser rastreada através de vários maneiras diferentes, com o uso de luvas com sensores eletro-mecânicos, dispositivos magnéticos de rastreamento, entre outros. Essas são consideradas soluções invasivas, onde o usuário precisa fazer uso de algum dispositivo extra, precisando muitas vezes *vestir* o dispositivo. Uma outra alternativa é utilizar-se apenas da visão computacional, sem a necessidade de nenhum dispositivo extra, ou seja, ter à disposição apenas as imagens provenientes de um dispositivo de captura de imagens, sendo assim uma solução não-invasiva.

A pesquisa feita neste trabalho se concentrou no estudo de algumas técnicas não-invasivas de rastreamento das mãos bem distintas, que fazem uso apenas da visão computacional para resolver o problema. Não usar nenhum dispositivo adicional foi um requisito básico para a pesquisa, lembrando que

a nossa intenção é aproximar a interação homem-máquina da interação natural entre homens. Uma vantagem disso é que, por utilizar-se apenas de um microcomputador e uma webcam comum, as soluções baseadas na visão computacional apresentam custo baixo, mais acessível. Outro requisito para a seleção das técnicas foi que estas apresentassem bons resultados em imagens com fundo heterogêneo.

Não foram pesquisados apenas os métodos de rastreamento das mãos, mas também técnicas voltadas para outras aplicações da visão computacional, como por exemplo técnicas de rastreamento de objetos em geral, visando ampliar os conhecimentos gerais de visão computacional.

Os experimentos foram feitos com o auxílio do pacote ImageJ, uma versão multiplataforma, ainda em desenvolvimento, do pacote NIH Image, para Macintosh. Além de ser um software livre com código aberto, este pacote tem como principal característica a disponibilidade de diversos algoritmos para manipular os mais variados formatos de imagens, melhorar a qualidade das imagens, realizar a detecção de bordas, operações morfológicas e diversos tipos de cálculos relacionados ao processamento de imagens, como o cálculo das áreas, médias, centróides, etc. É possível adicionar ainda mais recursos, além de todos os que já vêm embutidos no pacote, através de módulos escritos em Java. Um outro fator que ajuda muito é a existência de um grande número de programadores trabalhando no seu desenvolvimento, o que possibilita a contínua criação de novos módulos, conforme escrito em Pistori e Neto, 2003 [54].

## 1.4 Estrutura do Trabalho

No próximo capítulo fêz-se uma fundamentação teórica sobre o tema do trabalho, escrevendo sobre as interfaces homem-máquina, as mãos e gestos, e introduziu-se o processamento de imagens e o campo da visão computacional chamado de rastreamento visual.

No capítulo 3 foram tratadas e comentadas as técnicas de rastreamento das mãos estudadas no decorrer do trabalho.

No penúltimo capítulo escreveu-se sobre a Língua Brasileira de Sinais, o conversor LIBRAS-texto e, finalmente, a implementação do rastreamento das mãos no conversor.

Por fim, escreveu-se as considerações finais sobre o trabalho, com algumas idéias de trabalhos futuros.

# Capítulo 2

# Fundamentação Teórica

A análise dos gestos das mãos, baseada em imagens, é o meio mais natural para construção de interfaces homem-máquina baseadas em gestos, porém é a mais difícil de ser realizada. A utilização da visão computacional é a maneira não-invasiva de se fazer reconhecimento de gestos. Novas tecnologias permitem realizar o processamento de imagens em tempo real, o que têm tornado possível a interação homem-máquina baseada apenas na visão computacional. Neste capítulo trataremos primeiramente das interfaces homem-máquina e das vantagens e desafios da visão computacional aplicada a esta área. Em seguida, descreveremos a mão, seus pontos positivos e suas limitações, e definiremos e destacaremos as características e classificações dos gestos. Por último, descreveremos brevemente alguns conceitos de processamento de imagens, essenciais para um melhor entendimento das técnicas apresentadas nos capítulos seguintes.

## 2.1 Interface Homem-Máquina

Uma interface pode ser descrita como o meio pelo qual o usuário troca mensagens em alto nível com o computador. Essas mensagens podem ser de diferentes tipos, em ambas as direções, de acordo com o tipo de interface. Se é apenas textual, por exemplo, a interação homem-máquina vai ocorrer apenas através da digitação e da apresentação na tela dos comandos e das mensagens alfa-numéricas. Se é uma interface gráfica, as mensagens trocadas possuem muito mais detalhes, podendo correponder a cliques em botões, apresentações de caixas de diálogo, etc. Em baixo nível, entretanto, o que realmente importa é quão informativas são as mensagens que o sistema troca com o usuário, e é de responsabilidade do desenvolvedor construir uma interface que o usuário considere fácil de usar, amigável. Atualmente, pode ser

acrescentado ao conceito de interface qualquer parte do computador com a qual o usuário entra em contato, seja uma parte física, perceptiva ou conceitual.

De acordo com Porta, 2002 [57], interfaces gráficas podem ser classificadas em diferentes categorias de acordo com os tipos de entradas e saídas que elas recebem e disponibilizam:

- Interfaces multimídia são aquelas que disponibilizam para o usuário diferentes tipos de saídas (pelo menos dois tipos diferentes). Elas são focadas na mídia, podendo esta ser em forma de texto, imagens, sons, etc. A maioria das interfaces de usuário desenvolvidas recentemente exploram a multimídia de alguma forma (pelo menos combinando texto com imagens).

- Interfaces perceptivas são aquelas que tentam fazer com que o computador tenha capacidades perceptivas, e assim consiga capturar informações sobre o usuário e o seu ambiente. A máquina adquire a capacidade de ver, ouvir, etc. As interfaces baseadas na visão fazem parte desta categoria.

- Interfaces multimodais exploram as múltiplas formas de entradas e/ou saídas. Por exemplo, a entrada pode ser feita através do teclado, mouse, voz, gestos, etc. A saída multimodal não difere muito do conceito da saída multimídia. As interfaces multimídia podem ser vistas como um subconjunto das interfaces multimodais. Para diferenciar melhor, segundo Turk e Robertson, 2000 [74], multimídia tem como foco a mídia usada (imagens, sons, etc), enquanto a multimodalidade se concentra nos canais perceptivos humanos (visão, audição, etc).


Figura 2.1: As interfaces perceptuais combinam multimídia, multimodalidade e percepção para aproximar a interação homem-máquina da natualidade da interação entre pessoas

As três categorias de interface descritas acima, juntas, formam a classe de interfaces perceptuais (Figura 2.1). Interfaces perceptuais (*Perceptual*

*User Interfaces* ou PUIs) integram as características multimídia, perceptiva e multimodal para disponibilizar ao usuário meios de interação mais naturais e poderosos. Segundo Pentland, 2000 [51], as interfaces perceptuais têm como objetivo capacitar as máquinas a sentir o ambiente e a pessoa com a qual está interagindo, explorando a comunicação multimídia. As interfaces perceptuais representam o próximo passo na evolução da interação homem-máquina (Porta, 2002 [57]).

### 2.1.1 Interfaces baseadas na visão

Existem vários dispositivos de interação homem-máquina (figura 2.2), dentre os quais podemos citar os tradicionais: teclado, mouse, joystick e controles remotos. Existem também dispositivos mais sofisticados e menos comuns, como por exemplo, os dispositivos magnéticos (ex.: Polhemus Fastrack [1]), os dispositivos de interação 3D (ex.: SpaceBall [2]), e os dispositivos com sensores eletro-mecânicos (ex.: CyberGlove [3]).


Figura 2.2: Dispositivos de interação homem-máquina. a)Dispositivos comuns: teclado, mouse, joystick e controle remoto. b) Dispositivo com sensor eletro-mecânico: CyberGlove. c) Dispositivos de interação 3D: SpaceBall. d) Dispositivo magnético: Fastrack. e) Dispositivo de captura de imagem: WebCam

Muitos destes dispositivos são mais baratos, mais confiáveis e mais fáceis de serem configurados do que um sistema baseado na visão computacional. Então quais são as vantagens da visão computacional? O que nos faz confiar

[1]Informações sobre o Fastrack: `http://www.polhemus.com/fastrak.htm`

[2]Informações sobre o SpaceBall: `http://www.abs-tech.com/produtos/descontinuados/spaceball_4000_flx/spaceball_4000_flx.html`

[3]Informações sobre a CyberGlove: `http://www.simsol.co.uk/cyberglove.shtml`

que essa área vai ser amplamente utilizada como interação homem-máquina no futuro?

### 2.1.2 Vantagens da Visão Computacional Aplicada à Interação Homem-Máquina

Primeiramente, a visão computacional é um meio de entrada barato. Uma câmera digital pode ser integrada em um único chip e, pela sua simplicidade, já tem sido comercializada até mesmo integrada a outros equipamentos, como celulares e relógios. Por esse motivo também, a sua produção em massa é muito mais fácil de ser realizada do que de outros dispositivos de entrada com partes mecânicas, como por exemplo, as luvas de dados (CyberGlove). Além disso, os gastos com equipamentos para processamento de imagens podem ser economizados, pelo fato de que os processadores da maioria dos computadores atuais são rápidos o suficiente para tomar conta desta tarefa sozinhos.

Mais importante ainda, a visão computacional é versátil. Enquanto outros dispositivos de entrada, tais como o mouse, o teclado e o joystick, são limitados a funções específicas, a visão computacional pode ser utilizada em uma infinidade de aplicações. É importante lembrar que essas aplicações não são apenas da área de interação homem-máquina, visto que ela também é utilizada para autenticação, video-conferência, ensino a distância, etc.

A partir do nosso ponto-de-vista, a vantagem mais importante da visão computacional é ser não-invasiva. Similar aos microfones, câmeras são dispositivos abertos de entrada de dados, ou seja, não é necessário um contato direto com o usuário para capturar a entrada. Assim sendo, o usuário pode interagir com o computador livremente, sem nenhum fio ligado a ele, e sem a necessidade de manipular dispositivos intermediários. Por esse motivo nós focamos nosso trabalho nos algoritmos de visão computacional que não requerem nenhum equipamento junto ao corpo, como por exemplo, marcadores, luvas coloridas, etc. Tais marcadores simplificariam muitos problemas da visão computacional, porém destruíriam a sua principal vantagem: ser não-invasiva.

Aplicações fazendo uso de reconhecimento de expressão facial, rastreamento dos olhos e das mãos, reconhecimento de gestos, entre outros, eram considerados os problemas mais difíceis de serem resolvidos e menos propensos ao sucesso algumas décadas atrás (Pentland, 2000 [51]). Felizmente, esse paradigma mudou e, atualmente, exite uma proliferação muito grande de projetos de pesquisa em todo o mundo trabalhando no desenvolvimento de interfaces homem-máquina baseadas na visão. A visão computacional,

juntamente com o reconhecimento de voz, têm possibilitado que a naturalidade na interação entre homens esteja aparecendo também na interação homem-máquina.

Turk, 1998 [73], afirma que as interfaces baseadas na visão tentam responder às seguintes questões: *Tem alguém aí? Onde eles estão? Quem são eles? O que eles estão fazendo?* Isso quer dizer que as interfaces baseadas na visão tentam, automaticamente, capturar informações sobre o usuário e o seu ambiente, visando interpretar os seus comandos naturais feitos com o corpo, como por exemplo, reconhecimento de gestos (Ramamoorthya et al, 2003 [59]) ou da direção do olhar (Colombo e Bimbo, 1997 [14]).

Entre as várias utilidades que as interfaces baseadas na visão podem ter em aplicações, podemos citar as seguintes:

1. Rastreamento da cabeça: a posição da cabeça pode ser explorada para prover alguns tipos de entrada para o computador. Por exemplo, para ativar barras de rolagens em janelas ou para mudar o foco de uma janela para outra.

2. Reconhecimento da face ou da expressão facial: com o objetivo de personalização, em alguns sistemas, pode ser de grande utilidade identificar quem está à frente do computador (especialmente em operações cujo acesso é restrito). Além disso, distinguir entre diferentes expressões faciais pode ajudar a criação de uma interface mais *humana*, que consegue, por exemplo, reconhecer quando o usuário está com dificuldade em realizar alguma tarefa no sistema.

3. Rastreamento dos olhos: detectar a direção do olhar pode ser muito útil para usuários que tenham deficiências físicas que os impedem de usar o mouse. Ainda, identificação dos olhos pode ser um bom suporte para o rastreamento da cabeça e o reconhecimento da face ou expressão facial.

4. Reconhecimento de Gestos: posturas e movimentos das mãos e braços podem representar um meio de entrada de dados muito interessante para o computador, particularmente, quando as mensagens desejam expressar alguma forma de comunicação dependente de posição no espaço, ou mesmo tamanho. Complementando, os gestos podem ser usados como um tipo de alfabeto para formular comandos explícitos.

As funcionalidades alcançadas através de técnicas de visão computacional são basicamente as mesmas para os mais variados problemas, tais como o rastreamento da cabeça (figura 2.3), o reconhecimento de gestos, etc. Apesar de

cada tipo de problema requerer uma adaptação apropriada do método a ser utilizado para resolvê-lo, existem alguns métodos básicos que são utilizados independentemente do objeto a ser detectado. Por exemplo, as técnicas baseadas na aparência e as técnicas baseadas em modelos 3D podem ser usadas tanto para identificar posturas da mão como da cabeça. Do mesmo modo, o reconhecimento dos gestos das mãos e dos movimentos da cabeça podem ser solucionados baseando-se nas mesmas abordagens estatísticas, como por exemplo, os Modelos Ocultos de Markov.

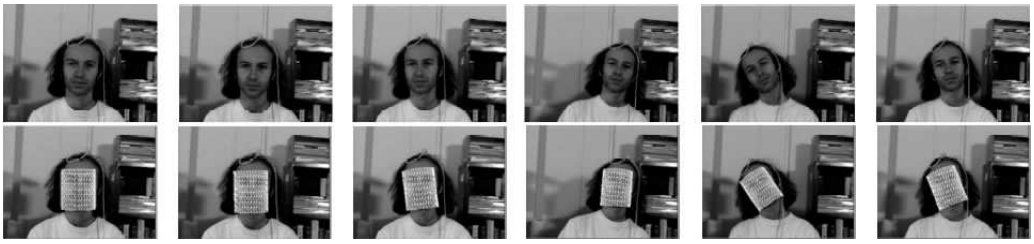

Figura 2.3: Imagem reproduzida de La Cascia et al, 1999 [35], exemplificando o rastreamento da cabeça.

De acordo com Turk, 1998 [73], na maioria das vezes, a função principal de uma interface baseada na visão se concentra em uma das duas possíveis categorias (não mutuamente exclusivas):

1. Controle: quando uma interface baseada na visão é utilizada para relacionar os movimentos que o usuário faz com o corpo, com as ações a serem executadas pelo sistema, ou seja, trata-se de um tradutor que interpreta os movimentos do corpo e os traduz para ações no sistema. Um exemplo: uma certa pose da mão pode indicar que o aplicativo atual deve ser fechado.

2. Consciência: quando uma interface baseada na visão é capaz, de forma autônoma, de perceber as necessidades do usuário através de seus atos e seu comportamento. Sem a comunicação explícita por parte do usuário, a interface reconhece certas situações e, sem a intervenção humana, decide as operações que devem ser executadas. Por exemplo, um sistema pode ser capaz de *ver* que o usuário está olhando para outro lugar quando uma falha em uma de suas tarefas ocorre e, para chamar a atenção visual do usuário de volta para o computador, emite um alerta sonoro. Bradshaw, 1997 [7], cita que as interfaces *conscientes* baseadas na visão, que sempre buscam a atenção visual do usuário, podem ser muito úteis para o desenvolvimento de sistemas que têm como objetivo auxiliar o usuário na interação com a máquina, também chamados de

> sistemas agentes. Lieberman, 1997 [38] complementa afirmando que os sistemas agentes irão explorar cada vez mais as interfaces baseadas em visão, em um futuro muito próximo.

A complexidade das interfaces baseadas na visão depende das aplicações onde serão utilizadas. Assim, as interfaces baseadas na visão podem ser, desde simples ferramentas que detectam quando existe alguém sentado em frente à tela do computador, até complexas implementações capazes de reconheces múltiplos comandos simultaneamente. É importante ressaltar que as implementações mais complexas muitas vezes precisam de dispositivos mais caros, como por exemplo, câmeras especias, hardware dedicado, etc. Esse paradigma vêm mudando a medida que os dispositivos de captura de imagem têm se tornado mais acessíveis, lembrando que hoje em dia é possível desenvolver muitas aplicações diferentes com interfaces baseadas na visão, fazendo uso de imagens capturadas das mais simples webcams disponíveis no mercado.

### 2.1.3 Desafios da Visão Computacional Aplicada à Interação Homem-Máquina

Muitos problemas de visão computacional, tais como a detecção e o rastreamento das mãos em frente a um fundo uniforme, parecem ser fáceis de serem resolvidos aos olhos humanos. Crianças podem realizar essa ação sem precisar de concentração. Mas o que parece simples para nós é, na realidade, o resultado de muito processamento realizado pelo cérebro.

De acordo com os livros de anatomia, a retina humana tem aproximadamente 125 milhões de células receptoras. Hardenberg, 2001 [23], cita que nós somos capazes de detectar e identificar quando existe um objeto no nosso campo visual a uma taxa de aproximadamente 25 hertz. Multiplicando esses valores, podemos concluir que, para que um computador apresente o mesmo desempenho, ele precisa ter pelo menos 3 gigahertz de processamento para cada instrução simples. Computadores com essa capacidade de processamento já estão disponíveis no mercado, com um custo nada acessível, mas a redução nos preços é apenas uma questão de tempo.

Mesmo se reduzirmos o número de valores de entrada para apenas 100.000 (384x288 pixels), o problema básico da computação continua óbvio: a grande quantidade de dados. Um computador não tem as mesmas capacidades de processamento paralelo que o cérebro apresenta. Devido a sua simplicidade, as operações são propensas a erros.

Outro grande problema das interfaces baseadas na visão é a falta de confiança e a falta de estabilidade dos resultados obtidos, causados princi-

palmente por mudanças nas condições de iluminação, oclusão, borramento causado por movimento rápido, e ruído elétrico. O sistema visual humano usa muitas dicas visuais em paralelo (cores, movimento, detecção de bordas), em combinação com conhecimentos de alto nível, para lidar com esta instabilidade.

Finalmente, existe sempre uma ambiguidade envolvida na interpretação de uma imagem. Por exemplo, a similaridade visual de uma mão e a sua sombra. Os homens, normalmente, não confiam apenas na saída da retina e, por isso, usam conhecimentos adquiridos para corrigir os erros e ambiguidades. Uma sombra é reconhecida através do uso inconciente do conhecimento sobre a posição 3D da mão e a fonte de luz, e sobre a propagação dos raios de luz no espaço. Para construir sistemas de visão computacional confiáveis, é necessário incorporar esses conhecimentos humanos aos algoritmos. No nosso caso, por exemplo, é interessante incluir informação sobre os formatos das mãos, a posição dos dedos em relação às mãos, etc. Por essa razão, até o presente momento, é impossível você construir um sistema genérico que trabalha com todos os tipos de objetos, em todos os tipos de ambientes.

## 2.2 Mãos: Dispositivos de Interação Homem-Máquina

Um gesto consciente é realizado quando a pessoa tem intenção em se comunicar ou completar uma tarefa (indicar, rejeitar, pegar, desenhar, etc), ou seja, é a expressão física de um conceito mental. Na comunicação entre homens, nós nos comunicamos através de palavras, e também fazemos uso do nosso corpo, das nossas mãos e dos nossos olhos. Nós utilizamos alguns movimentos para enfatizar uma idéia, sentimento ou atitude. Por exemplo, quando estamos em dúvida, balançamos os ombros, ou apontamos para algum lugar quando queremos explorá-lo. O ideal seria utilizar esses mesmos gestos naturais para interagirmos com os computadores. Nesta seção descreveremos primeiro a mão, seus pontos positivos e suas limitações. Em seguida definiremos e destacaremos as características e classificações dos gestos.

### 2.2.1 A Mão

A mão é um órgão do corpo humano muito adaptável. De acordo com Sturman, 1992 [70], essa adaptabilidade refere-se à facilidade com que as mãos são capazes de passar lenta ou rapidamente de uma função para outra. Além disso, a mão é um objeto altamente deformável, que possui todas essas qualidades devido à associação dos músculos com as várias articulações que co-

nectam os ossos ao esqueleto. A mão possui entre 27 e 29 graus de liberdade (Sturman, 1992 [70]; Rehg, 1995 [60]). A facilidade e naturalidade da sua utilização possibilita que a mão execute tarefas complexas em um caminho ótimo, lembrando que são as numerosas interconecções entre os músculos e os tendões que garantem essa grande complexidade ao movimento. A maior parte da força muscular vêm do ante-braço que, através de longos tendões, transmite a força para os dedos.

Segundo Marcel, 2002 [42], a mão possui algumas limitações nos movimentos, devido aos ângulos conhecidos de movimento das articulações. Os dedos executam movimentos de inflexão, extensão, abdução e adução (Figura 2.4). O polegar é desunido da palma da mão. Assim, os 3 graus de liberdade da articulação trapezoidometacarpal (situada na base do polegar) permite ao polegar realizar movimentos de rotação longitudinais, ao contrário dos outros dedos.

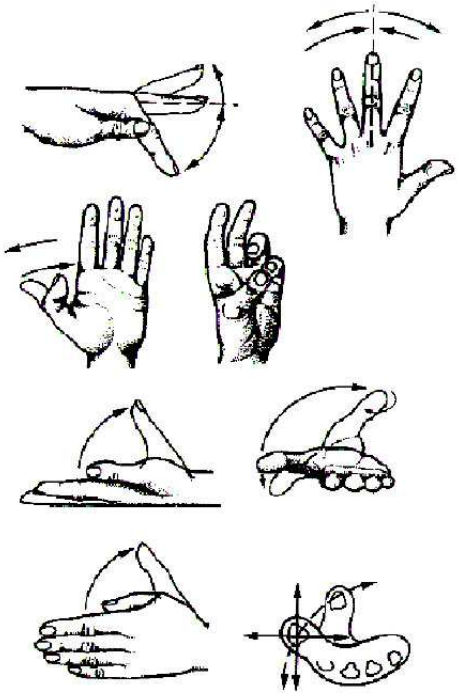

Figura 2.4: Imagem reproduzida de Marcel, 2002 [42], demonstrando os possíveis movimentos dos dedos de acordo com a limitação imposta pelas articulações.

### 2.2.2 Os Gestos

Existe uma infinidade de movimentos das mãos que realizamos no decorrer do dia-a-dia, como por exemplo: rezando (as duas mãos juntas e esticadas), expressando raiva (levantando a mão fechada), acusando (apontando com o dedo indicador), elogiando (polegar apontado pra cima), criticando (polegar apontado para baixo), acenando ou saudando alguém, apontando para alguma coisa, descrevendo o formato de um objeto (movimento das mãos representando o contorno do objeto), cumprimentando, apreciando (batendo palmas), jogando cartas (sinais com as mãos entre os parceiros), passando a sensação de tamanho de um objeto, ou distância entre duas cidades (mãos se aproximando ou se distanciando), conversando com a utilização de língua de sinais. De acordo com Marcel, 2002 [42], os gestos são intuitivos e alguns são universais, como por exemplo, gestos representando confirmação ou reprovamento, localização, tamanho de objetos, ou mesmo estimação de distância. Além disso, a utilização de gestos na interação homem-máquina remove a dificuldade de uso de dispositivos, pois a interação se torna praticamente natural.

A análise e o reconhecimento de gestos podem ser muito úteis em muitos campos de aplicações, como por exemplo:

- robótica (Cheng e Huang, 1991 [12]);
- realidade virtual (Pavlovic et al, 1996 [50]);
- interfaces multimodais, controladas por voz e gestos (Waibel e Vo, 1993 [77]; Cassell et al, 1999 [9]);
- análise de imagens de jogadores em uma partida de futebol, sequências coreográficas de dançarinos, etc (Campbell e Bobick, 1995 [8]; Ohno et al, 2000 [47]);
- tradução automátiva de línguas de sinais (Nam e Wohn, 1996 [45]; Pistori e Neto, 2004 [53]).

### 2.2.3 Comunicação através de Gestos

A comunicação não-verbal é realizada principalmente utilizando-se gestos das mãos. De acordo com Crowley e Coutaz, 1995 [16], as mãos possuem 3 funções, que ele considera complementares ou, de alguma forma, interligadas: a função epistêmica, a função ergótica e a função semiótica.

- Na função ergótica, a mão interage com o ambiente transformando-o, como por exemplo, movendo ou alterando a forma de um objeto. Modelar vasos de barro, limpar poeira, etc, é resultado de gestos ergóticos. Na interação homem-máquina, os gestos foram primeiramente explorados nesta função: digitando no teclado, movendo o mouse, clicando botões.

- Na função epistêmica, a mão é considerada um órgão de percepção. Ao mover sua mão sobre um objeto, você sente a sua estrutura, descobre de que material ele é feito, entre outras propriedades. A utilização da função epistêmica tem crescido principalmente no campo da realidade virtual, com o auxílio de dispositivos, tais como: luvas, ou mesmo roupas, providas de sensores e atuadores, transformando os movimentos em informações para o computador. O problema é que, para o usuário sentir a forma de um vaso de barro virtual, por exemplo, só mesmo através de roupas ou luvas com atuadores para passar informação do ambiente de volta para o usuário. A desvantagem é que, com esses dispositivos adicionais, o usuário fica *amarrado* ao computador, e não é possível resolver esse problema com visão computacional.

- Na função semiótica, a mão é um órgão que envia informações para o ambiente. Um gesto de despedida, a utilização da língua brasileira de sinais, gestos operacionais para controlar aviões no solo, e até mesmo o vulgar *dedo*, são exemplos de gestos semióticos. De acordo com Marcel, 2002 [42], é possível dividir mais ainda, distinguindo a função semiótica em 3 categorias de gestos humanos, em ordem crescente de expressividade:

  - Gestos de comando: usados para comunicação intencional, são gestos com sentido simples, e adaptados aos contextos no quais são utilizados. Também são utilizados para manipulação de objetos reais (robótica, manipulação remota) ou objetos virtuais (interfaces gráficas, realidade virtual). Um bom exemplo é o *Arrasta e Solta* (Iannizzotto et al, 2001 [28]), que imita a utilização do mouse, possibilitando a manipulação de objetos virtuais na tela.
  - Gestos co-verbais: servem para ilustrar e complementar a comunicação verbal. Na realidade, gestos e fala são combinados para transmitir muitas informações simultaneamente. Por exemplo, enquanto a pessoa fala, pode utilizar-se dos gestos para transmitir informação espacial.
  - Gestos de línguas de sinais (Pistori e Neto, 2003 [54]): formam

uma língua real e oficialmente reconhecida como meio legal de comunicação em alguns países, como por exemplo, a LIBRAS no Brasil. Os gestos de línguas de sinais são mais estruturados e complexos que os gestos naturais.

A realização de um gesto com a mão pode ser visto como um processo dinâmico, que segue um esquema no espaço e no tempo. Quek, 1994 [58] define esse esquema da seguinte forma:

- Gestos são movimentos em 3 fases: a preparação (movimentos preparatórios a partir da posição inicial), o núcleo (posturas da mão, trajetória do gesto)e a retração (o retorno do movimento para a inércia).
- As mãos possuem uma configuração particular durante a execução do movimento.
- Movimentos lentos entre as posições de inércia não são considerados gestos.
- Gestos estáticos (postura da mão) precisam de um período de tempo para serem reconhecidos.

### 2.2.4 Análise de Gestos Baseada em Imagens

A maior parte das técnicas de interação homem-máquina baseadas nas mãos requerem a estimação e o rastreamento da posição da mão a cada unidade de tempo. Segundo Bebis et al, 2002 [5], esses dados, provenientes da análise da imagem das mãos, são utilizados principalmente para esses dois propósitos:

1. Comunicação: corresponde à interpretação do formato da mão e do movimento (gestos). Trata-se essencialmente de um problema de reconhecimento de padrões onde o propósito principal é classificar as sequências de movimentos das mãos em um conjunto finito de gestos. Exemplos de aplicações com esse propósito são os reconhecedores de línguas de sinais, como o de Pistori e Neto, 2003 [54], e o de Huang et al, 1995 [27].
2. Manipulação: corresponde à interpretação do formato da mão e do movimento como uma ferramenta de manipulação, o que requer a medição de um conjunto de parâmetros relacionados aos aspectos físicos de cada tipo de movimento, para tomar decisões sobre a interação entre as mãos e os objetos virtuais. Um exemplo de aplicação de rastreamento das mãos com o propósito de manipulação foi desenvolvido em Lin et al,

2002 [39]. Trata-se de um sistema virtual de DJ [4] controlado através do rastreamento das mãos.

A análise dos gestos das mãos, baseada em imagens, é o meio mais natural para a construção de interfaces homem-máquina baseadas em gestos, porém é a mais difícil de ser realizada. Novas tecnologias permitem realizar o processamento de imagens em tempo real, o que têm tornado possível a interação homem-máquina baseada apenas na visão computacional. Entretanto, essa abordagem traz algumas dificuldades que descreveremos no decorrer desta subseção: segmentar a mão na imagem, analisar as posturas e rastrear o objeto segmentafo na sequência de imagens.

Segundo Lu et al, 2002 [40], exitem vários dispositivos baseados em luvas com sensores eletro-mecânicos (Figura 2.5), que capturam os movimentos das mãos diretamente, no entanto esses sensores são caros e difíceis de serem utilizados. O rastreamento das mãos baseado na visão é uma alternativa barata e não-invasiva. Além disso, Iannizzotto et al, 2001 [28] cita que as técnicas de rastreamento baseadas apenas na visão computacional estão sendo as mais utilizadas, já que não requerem nenhum dispositivo adicional (luvas, etc.), e podem ser implementadas com imagens capturadas de uma simples webcam, tornando o seu baixo custo outro grande atrativo. Algumas técnicas fazem uso de luvas coloridas para facilitar o rastreamento das mãos na imagem, como em Bebis et al, 2002 [5]. Outras utilizam hardware específico, podendo-se citar como exemplo o *FingerMouse*, um sistema com duas câmeras e um sensor de distância, sempre visando obter um melhor resultado, conforme Hamette et al, 2002 [22].


Figura 2.5: Soluções invasivas: luvas com sensores eletromecânicos (capturam os movimentos das mãos diretamente) e luvas marcadas (facilitam o rastreamento das mãos).

[4]DJ: responsável por controlar, reproduzir, manipular músicas ou efeitos sonoros

De acordo com Crowley et al, 2000 [17], para o desenvolvimento de uma interface baseada em gestos, as funcionalidades presentes são as seguintes:

1. Detecção (Segmentação): seu propósito é determinar a presença ou ausência de algum objeto ou acontecimento de um certo tipo, separando-o do fundo da imagem. Nessa fase, as interfaces baseadas na visão tentam responder algumas questões, como por exemplo, *Existe uma mão na imagem? Ela se moveu?* Hardenberg, 2001 [23], cita que as técnicas simples de detecção são baseadas em cores, análise de movimento e de componentes conectados. Para simplificar a segmentação, de acordo com Marcel, 2002 [42], alguns sistemas utilizam marcadores passivos (pontos coloridos), marcadores ativos (diodos luminosos) ou a cor da pele. Como a intenção é deixar a interação o mais natural possível, não comentaremos sobre tais técnicas, ou seja, trataremos apenas de técnicas de segmentação que não fazem uso de nenhum dispositivo adicional. Segundo Hardenberg, 2001 [23], se conseguirmos realizar uma segmentação realmente eficiente entre o objeto de interesse (mão) e o fundo da imagem, o rastreamento se torna uma atividade extremamente fácil. Sendo assim, a partir deste momento, trataremos as técnicas de segmentação como também técnicas de rastreamento das mãos.

2. Identificação (Análise de Posturas): nesse estágio, a interface baseada na visão procura reconhecer um objeto ou acontecimento em particular. Por exemplo, tenta entender um gesto específico que o usuário está realizando, ou mesmo para onde ele está olhando. Um exemplo: identificação de símbolos escritos em um quadro-branco (Stafford-Fraser, 1996 [68]). Uma técnica muito usada para analisar as posturas e gestos é a reconstrução em 3 dimensões. Um modelo 3D de mão é combinado com uma ou mais imagens para estimar os parâmetros (orientações, ângulos das articulações). Entretanto, o problema de oclusão limita esta técnica quando usada com apenas um dispositivo de captura. Utilizando-se mais dispositivos, esse problema é amenizado, mas não resolvido, visto que ainda poderão existir situações de oclusão. Os modelos usados em 3 dimensões podem ser volumétricos ou em forma de esqueleto. Alguns modelos volumétricos, como o proposto em Wren et al, 1997 [80], e o proposto em Azerbayejani et al, 1996 [3], são formados por estruturas geométricas, tais como cilindros, esferas e elipses, que representam as mãos e outras partes do corpo. Esse método é muito custoso, e muito difícil de ser implementado em tempo real. Além disso, o conhecimento da postura exata da mão não tem utilidade, por exemplo, para gestos de comunicação, mas serve muito bem

para gestos de manipulação. Modelos de esqueleto, como o proposto em Ahmad, 1995 [2], são baseados nas características biomecânicas e morfológicas para representar os segmentos, com suas articulações e respectivos ângulos. Essa funcionalidade é muito útil em alguns sistemas, como por exemplo, na identificação de gestos representando letras de uma língua de sinais (Pistori e Neto [53]).

3. Rastreamento: uma vez que o acontecimento ou o objeto foi detectado e identificado, é necessário seguí-lo, rastreando continuamente a sua posição até que não seja mais necessário. Uma solução para isso seria realizar o processo de segmentação a cada nova imagem capturada, mas isso não é possível quando a tarefa de segmentação é muito difícil, como por exemplo, reconhecimento de faces, pois nesse caso o processo demora muito, não podendo ser executado em tempo real. O ideal é lembrar das últimas posições conhecidas e identificadas do objeto. Dadas algumas condições em relação aos possíveis movimentos do objeto entre 2 imagens consecutivas, o algortimo de rastreamento tenta seguí-lo no decorrer do tempo. Um exemplo: a interface baseada na visão pode estar interessada em saber onde está a mão neste momento. Conforme Marcel, 2002 [42], os gestos são processos tanto temporais como espaciais. Isso quer dizer que precisamos saber, a cada intervalo de tempo, ou a cada nova imagem capturada, a posição da mão previamente reconhecida, para estabelecer a trajetória do movimento. Existem várias técnicas para predizer a próxima localização da mão, ou mesmo para estimar os parâmetros de um modelo 2D ou 3D, usando um modelo dinâmico (Rehg e Kanade, 1993 [61]) ou filtros de Kalman (Vaillant e Darmos, 1995 [76]), entre outros. Como o rastreamento das mãos é o tema central deste trabalho, descreveremos várias técnicas, incluindo a nossa implementação, nos próximos capítulos.

## 2.3 Fundamentos de Processamento de Imagens

Nesta seção descreveremos brevemente os conceitos essenciais para um melhor entendimento das técnicas apresentadas nos capítulos seguintes. Para realizar um estudo mais profundo dos fundamentos de processamento de imagens, o leitor deve procurar por livros da área, tais como: Gonzalez e Woods, 1992 [20]; Jain, 1989 [32]; Umbaugen, 1998 [75], os quais foram utilizados como base para escrever esta seção. Esses são só alguns exemplos de livros, visto que a bibliografia na área é muito extensa.

### 2.3.1 Imagens Digitais

Basicamente, uma imagem digital pode ser vista como uma matriz de elementos chamados *pixels*. Quanto maior a quantidade de pixels, maior a resolução da imagem, ou seja, mais detalhada é a imagem. Existem dispositivos, como os celulares providos de câmeras e as webcams, que capturam imagens pequenas, geralmente entre 160x120 e 352x288. Assim como também existem câmeras profissionais que capturam imagens em alta resolução, geralmente com a finalidade de impressão, chegando a capturar imagens de 1600x1200 pixels ou até maiores. As imagens são capturadas pelas câmeras à uma taxa, chamada de número de imagens por segundo ou *frames* por segundo (fps), que depende da aplicação do equipamento. Quando é necessário uma estimação mais precisa de profundidade da cena, é comum a utilização de duas ou mais câmeras (visão estérea).

Em imagens em níveis de cinza (ou escalas de cinza, ou tons de cinza), cada pixel tem um valor associado, que indica a intensidade de sua luminância. Usando 8 bits por pixel, por exemplo, os valores podem variar de 0 (preto) até 255 (branco). Em imagens coloridas RGB [5], cada pixel possui 3 valores associados, sendo um para a intensidade da cor vermelha, outro da cor verde e outro do azul. Qualquer cor pode ser obtida com uma combinação apropriada dessas 3 cores-base. Supondo que sejam atribuídos 8 bits para cada um dos 3 canais, podendo assim variar a intensidade de cada cor de 0 a 255, cada pixel terá um valor de 24 bits correspondendo a sua cor. O modelo de cores RGB não é o único disponível mas, no geral, é o mais apropriado para processamento básico de imagens, além de ser o modelo de cores padrão da maioria dos dispositivos de captura comercializados. É importante lembrar que, tanto nas imagens em níveis de cinza como nas imagens coloridas, quanto maior o número de bits de intensidade/cor por pixel, maior a qualidade da imagem.

### 2.3.2 Análise de Imagens

O *histograma* de imagens em níveis de cinza é um vetor cuja quantidade de elementos é igual ao número de valores possíveis por pixel (Figura 2.6). Cada elemento $i$ do vetor contém o número de vezes que o valor $i$ aparece na imagem. Considerando, por exemplo, uma imagem de 8 bits, seus valores podem variar de 0 a 255, ou seja, o vetor possui 256 valores. Assim, por exemplo, o elemento 77 do vetor vai conter o número de pixels cujo valor (intensidade) seja igual a 77. Para as imagens coloridas, a idéia de histograma

[5] O nome do modelo de cores RGB vêm do nome em inglês das 3 cores-base utilizadas: Red (vermelho), Green (verde) e Blue (azul).

também é válida. O que muda é que, como a imagem é formada por 3 canais de cores, ela possui então 3 histogramas diferentes, um para cada cor básica. Tanto para imagens em níveis de cinza, como para imagens coloridas, os histogramas são muito úteis para caracterizar uma imagem, ou partes dela. As distribuições no histograma, de pequenas áreas de uma imagem, podem ser exploradas para, por exemplo, encontrar similaridades, ou correlações entre certas zonas de imagens distintas, capturadas pelo dispositivo.

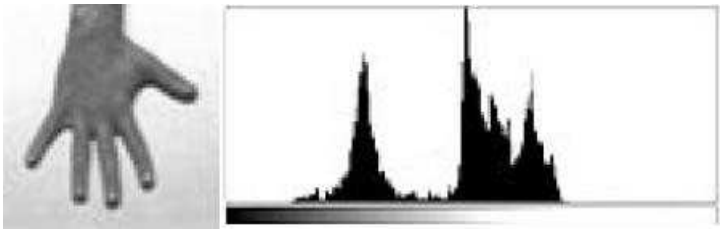

Figura 2.6: Histograma da imagem da mão.

A *segmentação* é a subdivisão de uma imagem em componentes, de acordo com algum critério de homogeneidade (morfológico, cromático, etc.). Geralmente, a segmentação serve para fazer a separação entre os objetos de interesse e o fundo da imagem, produzindo assim uma imagem binária, na qual cada pixel é classificado como pertencente ao fundo da imagem ou a um objeto de interesse, sendo cada uma das duas classes comumente associada a uma cor: branco ou preto (Figura 2.7).

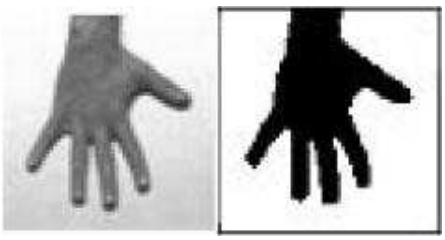

Figura 2.7: Segmentação da imagem da mão.

O processo de segmentação pode ser iniciado com uma busca por descontinuidades (variação abrupta de cor ou luminância) na imagem, que normalmente corresponde às bordas do objeto. Na prática isso pode ser feito com a utilização de aproximações discretas do gradiente da imagem, na forma de *máscaras de convolução*. Considerando, por exemplo, imagens em níveis de cinza, uma máscara de convolução é uma matriz $n + m$ que se move sobre a imagem para produzir uma imagem transformada. A cada iteração, o centro da matriz é posicionado em um pixel da imagem, e este pixel tem o seu valor

de intensidade alterado para a soma dos produtos dos valores de cada célula pela intensidade do pixel correspondente. Com a utilização de um *limiar*, a imagem pode ser então binarizada da seguinte forma: todos os pixels com intensidade maior que esse limiar são considerados parte da borda do objeto atual. Máscaras de convolução são também usadas na implementação de filtros genéricos, que possuem como propósito processar a imagem de alguma maneira, como por exemplo, eliminar algum ruído.

A segmentação pode ser executada também nas imagens coloridas, realizando uma busca por regiões homogêneas da imagem, ou seja, que possuam as mesmas características cromáticas. Por exemplo, veremos em uma das técnicas de segmentação que a segmentação baseada na cor da pele é muito utilizada para separar as mãos do fundo da imagem.

### 2.3.3 Estimação de Movimento

Dado um objeto em movimento na cena, o problema básico é seguir o seu movimento.

Utilizando-se de visão monocular (apenas um dispositivo de captura de imagem), a técnica básica para estimação de movimento é o *fluxo ótico*, formalmente definido como o campo vetorial da variação da luminância. O princípio utilizado é bem intuitivo: se um certo pixel tem valor de luminância *x1* em uma imagem capturada, e o mesmo pixel apresenta um valor $x2 \neq x1$ na imagem seguinte, significa que alguma coisa mudou na cena, possivelmente devido a movimento. Este método possibilita estimar a velocidade de movimento de cada pixel, desde que as duas condições a seguir sejam levadas em consideração:

1. o brilho da cena não deve mudar e
2. o ponto examinado(pixel) deve ser visível em ambas as imagens.

Apesar dessas condições não poderem ser cumpridas em algumas situações, o método do fluxo ótico (normalmente melhorado de alguma maneira) é quase sempre a base para encontrar objetos em movimento a partir de visão monocular.

## 2.4 Rastreamento Visual

O rastreamento de objetos em imagens de vídeos têm sido um tópico popular no campo da visão computacional. O rastreamento é uma função importante na análise de movimento humano, visto que ele prepara os dados para a estimação de poses e o reconhecimento de ações. De acordo com Wang et al,

2003 [78], contrastando com a segmentação, o rastreamento é considerado um problema de visão computacional de alto nível. Entretanto, os algoritmos de rastreamento das mãos muitas vezes apresentam uma intersecção considerável com os algoritmos de segmentação, durante o processamento, sendo que em alguns casos as técnicas de segmentação chegam a ser confundidas e tratadas como de rastreamento, devido aos resultados apresentados. Conforme escrito em Mini e Campos, 1999 [44], a habilidade de detecção de movimentos é uma parte fundamental do sistema de visão dos animais. Fazer com que as máquinas possuam capacidade semelhante é um desafio que vem recebendo maior atenção a cada dia.

Existem várias linhas de pesquisa nesta área, dentre as quais podemos citar muitas que utilizam uma sequência de imagens para detectar e acompanhar os objetos-alvo nas cenas, ação também conhecida como rastreamento visual de objetos (*tracking*). O rastreamento pode ser feito através de muitas maneiras. Em Mini e Campos, 1999 [44], é escrito que algumas se baseiam exclusivamente em regiões da imagem, enquanto outras, com informações prévias sobre os objetos a serem rastreados, constróem modelos matemáticos para serem utilizados no rastreamento. A base do problema consiste em determinar quais os objetos da cena serão acompanhados e então localizar cada um deles nos diversos quadros capturados. Trata-se de uma tarefa complicada, na qual podemos citar alguns problemas, como por exemplo: se o objeto rastreado não for rígido (uma mão por exemplo), conforme ele se movimenta, a sua forma se altera, o que dificulta o trabalho de seguí-lo ao longo dos diversos quadros. Outros problemas: diferenças de luminosidade durante o processo de rastreamento, existência de mais de um objeto em cena ocasionando a oclusão parcial ou total do objeto-alvo, ou mesmo quando a câmera não se encontra estática no ambiente monitorado, obrigando que se desconte seu movimento no processo de acompanhamento do objeto. De acordo com Wang et al, 2003 [78], o rastreamento pode ser dividido em diversas categorias, de acordo com diferentes critérios, apresentados a seguir.

- Partes humanas rastreadas: mãos, rostos, olhos, pernas, o corpo todo, etc.
- Número de partes humanas rastreadas: uma mão apenas, ou duas, uma pessoa, múltiplas pessoas, etc.
- Número de imagens da cena: visão simples, visão múltipla, e visão omnidirecional.
- Dimensão do espaço de rastreamento: 2D, 3D, etc.
- Ambiente de rastreamento: interno ou externo.

- Estado da câmera: em movimento, estática.

Partindo dessas categorias, definimos as técnicas estudadas nesse trabalho da seguinte forma: rastreamento de uma mão, através da visão de apenas um dispositivo de captura (visão simples), no espaço 2D e em um ambiente indoor, com um dispositivo de captura estático.

Dois fatores importantes, que sempre devem ser levados em consideração nas técnicas de rastreamento visual, são a velocidade e a robustez. De acordo com Grassi, 2002 [21], velocidade pode ser definida como sendo o tempo ou a frequência com que o método pode ser executado, e robustez pode ser definida como sendo a capacidade de um sistema de rastrear de forma precisa mesmo quando as circunstâncias visuais não são favoráveis ou ideais. De uma forma geral, existe uma relação de compromisso entre velocidade e robustez, pois quanto menor o tamanho da janela que limita a busca do alvo dentro da imagem, menor a robustez do sistema de rastreamento e maior a velocidade de processamento. Assim sendo, deve haver um balanceamento entre esses dois fatores, atentando sempre para o fato de que a maioria dos sistemas de rastreamento possuem como necessidade o processamento em tempo real.

Em Zwaan, Santos-Victor, 2001 [84], são mostrados alguns resultados obtidos através da seguinte análise: aumentando a dimensão da janela de busca ocorre a diminuição do erro (maior robustez) e, consequentemente, a diminuição da frequência de rastreamento (menor velocidade).

Já em Toyama e Hager, 1999 [71] é proposto um sistema de rastreamento visual em tempo real com uma arquitetura robusta e adaptativa, na qual são utilizados vários métodos de rastreamento visual dispostos em camadas. Os métodos são classificados por velocidade e robustez, e o sistema controla a transição entre eles de forma a executar o método mais adequado de acordo com as condições visuais do ambiente no qual o objeto-alvo está inserido.

Entre as ferramentas matemáticas úteis para rastreamento, podemos citar os filtros de Kalman(Welsh e Bishop, 1995 [79]), o algoritmo de condensação (*Condensation* - Isard e Blake, 1998 [30]; Sidenbladh et al, 2000 [66]), as redes Bayesianas dinâmicas (Pavlovic et al, 1999 [49]), etc. O método de Filtros de Kalman é uma estimação de estados baseada em distribuções Gaussianas. A desvantagem desses filtros é que eles são inadequados para lidar com situações nas quais haja a presença de oclusões, fundos heterogêneos com objetos semelhantes aos objetos rastreados, etc, visto que este tipo de situação acaba gerando distribuições multimodais, e os filtros de Kalman só trabalham com distribuições unimodais. De acordo com Isard e Blake, 1998 [30], o algoritmo de condensação é um método de propagação condicional de densidades para o rastreamento visual, e têm se mostrado uma alternativa poderosa. O algoritmo usa uma técnica chamada de *amostragem fatorada*, que anteriormente

era aplicada na interpretação de imagens estáticas, na qual a distribuição de probabilidade das possíveis interpretações é representada por um conjunto gerado aleatoriamente. Ele usa alguns modelos dinâmicos, juntamente com observações visuais, para propagar o conjunto aleatório através do tempo, resultando em um rastreamento altamente robusto de movimento que pode ser processado muito próximo do tempo real.

# Capítulo 3

# Rastreamento das Mãos

O nosso foco é discutir vários métodos utilizados no processo de rastreamento. Hardenberg, 2001 [23], cita que, se conseguirmos realizar uma segmentação realmente eficiente entre o objeto de interesse (mão) e o fundo da imagem, o rastreamento se torna uma atividade extremamente fácil. Por esse motivo, dentre as técnicas apresentadas, existem algumas que são classificadas como técnicas de segmentação, e não de rastreamento, mas que, se utilizadas com eficiência, acabam realizando também o papel da etapa de rastreamento. É importante ressaltar que alguns dos trabalhos citados são de rastreamento do corpo humano como um todo, e não especificamente das mãos. Pelo fato do corpo humano ser classificado como um objeto não-rígido, assim como as mãos, os trabalhos citados são válidos. A extensa bibliografia relacionada a este domínio provê inúmeras técnicas diferentes. Nesta seção iremos detalhar alguns dos principais métodos de rastreamento já publicados.

## 3.1 Rastreamento Baseado em Modelos

A estrutura das mãos pode ser representada por contornos 2D ou modelos volumétricos 3D. A escolha do modelo depende da aplicação do rastreador. Se a saída do rastreador vai ser utilizada para algum processo de reconhecimento, um modelo 2D já é suficiente. Por outro lado, um modelo 3D pode ser necessário quando a aplicação requer maiores detalhes do rastreamento. Essas são alternativas muito utilizadas, sobre as quais escreveremos um pouco a seguir.

### 3.1.1 Contornos 2D

Nesse tipo de representação, as partes do corpo são diretamente relacionadas com a projeção do corpo no plano da imagem. Por exemplo, em Isard e MacCormick, 2000 [31], foi apresentado um sistema de rastreamento, baseado na visão computacional, que trabalha confiavelmente em tempo real, e apresenta bons resultados, mesmo em fundos heterogêneos e com movimentos rápidos das mãos. Depois de encontrada a mão, através da busca pelos contornos, feita pelo algoritmo de condensação, o dedo indicador é utilizado para desenhar na tela. Estender o polegar gera cliques de mouse, e o ângulo do polegar em relação à mão controla a espessura da linha de desenho (Figura 3.1).

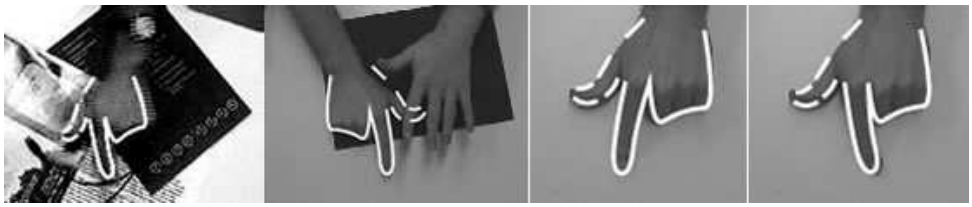

Figura 3.1: Resultados de Isard e MacCormick, 2000 [31], mostrando que a solução deles funciona muito bem em fundos heterôgêneos.

Extrair o contorno ou silhueta tanto do modelo como da imagem é uma tarefa relativamente fácil. Em Cipolla e Hollinghurst, 1996 [13], é apresentado um sistema de rastreamento que usa um modelo 2D deformável, criado através de transformações que preservam as linhas e o paralelismo dos objetos na imagem, também chamadas de transformações afins.

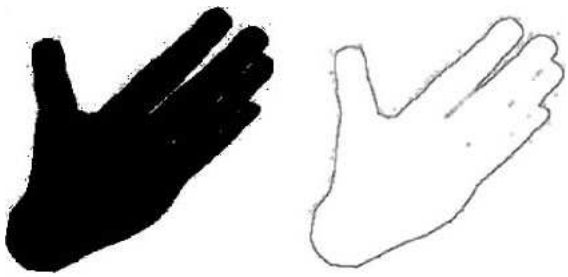

Figura 3.2: Silhueta e contorno 2D.

### 3.1.2 Modelos Volumétricos

A desvantagem de modelos 2D é a sua restrição em relação ao ângulo do dispositivo de captura. Por isso, muitos pesquisadores vêm tentando descre-

ver a estrutura geométrica das mãos em maiores detalhes utilizando modelos 3D, tais como: cilindros, cones, esferas, etc. Quanto mais completo o modelo 3D, maior a robustez obtida, mas isso acaba prejudicando a velocidade do processamento, levando a computações muito custosas durante o processo de combinação entre o modelo e a imagem.

Rehg e Kanade, 1994 [62], introduz o uso de um modelo de mão 3D altamente articulado para o rastreamento da mão humana (Figura 3.3). O modelo 3D é formado através de inúmeros cilindros representando as falanges dos dedos. A posição dos dedos é rastreada através da combinação das bordas (obtidas através do filtro de Canny). É aplicado um método dos mínimos quadrados não-lineares para minimizar os erros entre a localização das junções e das falanges dos dedos. O sistema funciona em tempo real, entretanto não apresenta bons resultados quando se defronta com oclusões ou mesmo com um fundo muito heterogêneo.

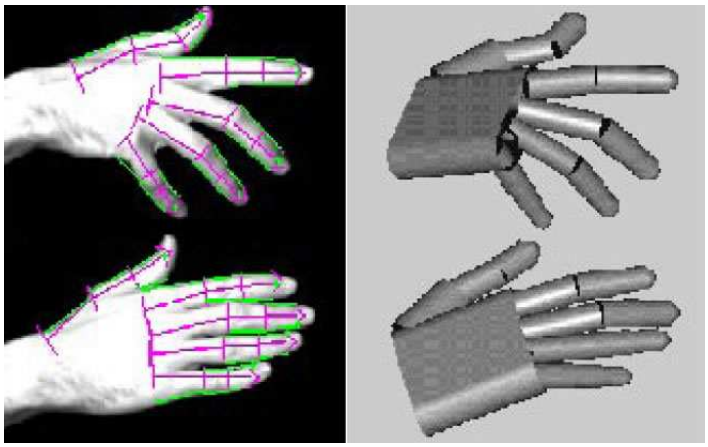

Figura 3.3: Resultados de Rehg e Kanade, 1994 [62]: primeiro a imagem original, com um esqueleto sobreposto e, depois, o modelo de mão 3D derivado dos parâmetros extraídos.

Já em Heap e Hogg, 1996 [25], é utilizado um modelo 3D deformável. A mão é modelada como uma superfície de colméia, e é construída através dos resultados de uma rede PCA (Principal Component Analysis, descrita em Fiori e Piazza, 1999 [18]) treinada a partir de um número muito grande de imagens de mãos humanas. O rastreamento em tempo real é alcançado através da combinação entre o modelo de mão deformado que mais se aproxima da imagem da mão. A dificuldade desse método é a captura do número grande de imagens. Se a iluminação muda muito, por exemplo, um novo conjunto de imagens com as novas características de iluminação deve ser capturado, e a rede necessita incorporar esse novo conjunto, para só então o

sistema voltar a responder corretamente às imagens.

Segundo Wang et al, 2003 [78], uma vantagem importante dessa técncia é que, com a utilização dos modelos 3D é possível obter bons resultados na presença de oclusão, e também de obter dados mais significativos em relação a análise da ação em si. Entretanto, os modelos são restritos por suposições não muito práticas, dos limites de movimento da mão. Por esse motivo, em Hardenberger e Bérard, 2001 [24], toda vez que é tratado dos movimentos das mãos, é destacado que esses movimentos são para *pessoas que não tocam piano*, querendo dizer que quem toca piano realiza movimentos diferentes dos limites impostos. Além disso, a complexidade computacional é muito alta.

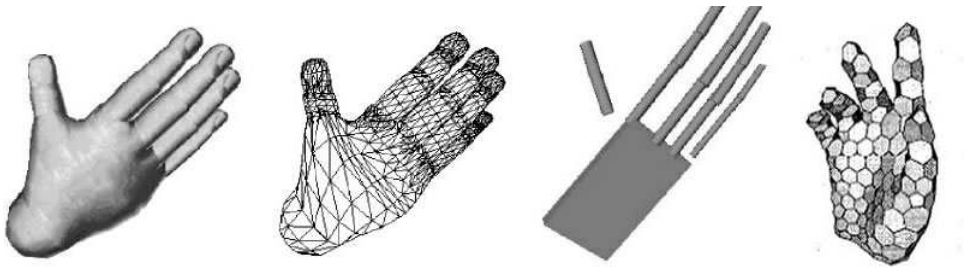

Figura 3.4: Modelos diferentes de mãos são utilizados no rastreamento, tais como, da esquerda para a direita: modelo volumétrico texturizado 3D, modelo volumétrico aramado 3D, modelo esqueletal 3D e modelo deformável 3D em forma de colméia.

## 3.2 Rastreamento Baseado em Regiões

Segundo Wren e Pentland, 1999 [81], o rastreamento baseado em regiões têm sido muito utilizado hoje em dia. A idéia desta abordagem é identificar uma região conectada para cada objeto em movimento na imagem, e assim rastreá-lo através do tempo usando uma medida de correlação cruzada (McKenna et al, 2000 [43]; Wren et al, 1997 [80]), ou alguma outra técnica de segmentação.

A técnica de rastreamento baseado em regiões funciona tentando agrupar as regiões homogêneas conectadas da imagem. Em relação à limiarização pura, o crescimento de regiões apresenta duas vantagens. Primeiramente, os valores dos parâmetros que definem a homogeneidade entre as regiões é calculado para cada região, e é atualizado cada vez que um novo pixel é adicionado a região, ou seja, o limiar é escolhido localmente. O outro ponto a ser destacado é que o algoritmo segmenta várias regiões na imagem, que podem ser processadas independentemente. Isso é muito bom no caso de existirem, por exemplo, duas mãos interagindo na imagem.

Existem duas variantes básicas desse algoritmo, que desde os anos 70 têm recebido grande atenção por parte dos pesquisadores em todo o mundo. Uma delas é a técnica de semear (*seeding*), descrita por Adams e Bischof, 1994 [1], onde um ou mais pontos na imagem são marcados como *sementes*, e a partir dos pixels vizinhos desses pontos vão se formando as regiões homogêneas. A outra variante muito difundida é a separação-e-junção (*split-and-merge*), descrita por Chen et al, 1991 [11], que segue o caminho contrário da anterior, começando com a imagem inteira e dividindo a mesma iterativamente, até restarem apenas regiões homogêneas. Em ambas as variantes, os algoritmos dependem muito do critério de homogeneidade utilizado.

## 3.3 Rastreamento Baseado na Cor-da-Pele

Para realizar o rastreamento das mãos, uma característica frequentemente utilizada é a cor, no nosso caso, a chamada *cor-da-pele*. O método mais utilizado é bem direto: alguns pixels do objeto são utilizados para construir um modelo de cores. Com um bom modelo, todos os pixels das imagens seguintes podem ser classificados rapidamente como sendo parte do objeto de interesse ou não. Depois de segmentado, o rastreamento torna-se uma tarefa trivial.

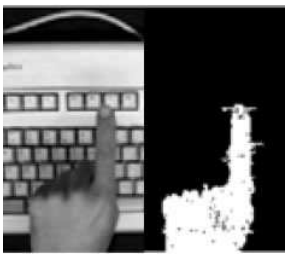

Figura 3.5: Demonstração de segmentação por cores.

Para melhorar a robustez da segmentação por cores, é necessário construir modelos de cores mais generalizados, a partir dos valores dos pixels amostrados. Vamos explicar a seguir três possíveis métodos que melhoram essa maneira de segmentar:

1. Transformação no espaço de cores: em aplicações reais, a iluminação pode mudar no decorrer do tempo. Nuvens passam na frente do sol, pessoas se movem na frente de fontes de luz, e o brilho de um objeto também pode mudar se ele se move no decorrer das cenas. Para alcançar estabilidade no decorrer do tempo, um modelo de cores generalizado deveria ser invariante em relação às mudanças nas condições

de iluminação. Segundo Schiele e Waibel, 1995 [64], cada valor de cor deve ser normalizado de acordo com a luminosidade, para assim obter um modelo invariante ao problema da iluminação. Uma outra alternativa é a transformação dos valores dos pixels amostrados, do espaço de cores RGB para o espaço Matiz-Luminância-Saturação, conhecido como HLS, que apresenta robustez ainda maior por permitir uma certa variação na saturação e na cor.

2. Modelo Gaussiano: Hardenberg e Bérard, 2001 [24], descrevem uma alternativa usando funções Gaussianas 2-D. Neste método, os pixels são normalizados pelas suas luminosidades, reduzindo assim a complexidade dos dados de entrada para duas dimensões. Em seguida, assume-se que a distribuição das cores presentes na pele humana pode ser aproximada pela curva Gaussiana 2-D do espaço de cores normalizado. Os valores dos pixels amostrados são usados para calcular os parâmetros dessa função (média e matriz de covariância). Uma vez construída, a função calcula uma probabilidade para cada possível valor de cor. Essa probabilidade é uma medida muito útil quando queremos ter a certeza de que o pixel pertence ao objeto (mão).

3. Segmentação com Teoria de Decisão Bayesiana: Kulessa e Hoch, 1998 [34], e posteriormente Zhu, Yang e Waibel, 2000 [83], descrevem uma estratégia de segmentação de cores baseada na teoria de decisão Bayesiana. Os valores das probabilidades utilizadas por essa técnica são extraídos através de um aprendizado supervisionado: um número grande de imagens, com as coordenadas do objeto segmentado em cada imagem, são apresentados. Segundo Hardenberg e Bérard, 2001 [24], a grande vantagem desse método é a robustez apresentada em casos em que as cores do fundo se assimilam com as cores do objeto. A desvantagem é que a performance do sistema é muito dependente da qualidade e da quantidade das imagens amostradas para a fase de treinamento. Se o fundo muda, ou se as condições de iluminação mudam, é necessário retreinar tudo para voltar a ter bons resultados. Zhu, Yang e Waibel, 2000 [83], propõem um algoritmo de maximização de expectativa (EM), com o objetivo de adaptar o modelo de cores dinamicamente. Uma outra alternativa é a utilização de mapas auto-organizáveis (SOM), descrito por Wu e Huang, 2000 [82].

O tempo de processamento de algoritmos que realizam a segmentação por cores é rápido, posibilitando a realização do rastreamento por cores em imagens de até 768x576 pixels em tempo real (Hardenberg, 2001 [23]). Em ambientes heterogêneos, fica muito difícil a construção de um modelo de cores

robusto, visto que a cor da pele não é sólida, variando de pessoa para pessoa, e as mudanças na iluminação também influem bastante. É possível porém obter bons resultados na segmentação apenas em ambientes com iluminação controlada, e com um conjunto bom de imagens amostradas.

## 3.4 Rastreamento Baseado na Correlação

Correlação é o nome dado à técnica de reconhecimento de padrões que mede a similaridade entre um padrão amostrado e qualquer outro padrão de teste. Para ser utilizado no rastreamento das mãos, o princípio básico da correlação é muito simples: uma imagem do objeto é amostrada no início, e então o algoritmo passa a buscar aquele objeto em todas as imagens seguintes, testando a imagem-padrão (a imagem do objeto de interesse) em todas as possíveis posições na imagem, ou seja, aplicando o algoritmo de correlação em todas as possíveis posições.

Para realizar o cálculo que mede a similaridade entre duas imagens, existem várias maneiras. Uma delas, descrita e avaliada dentre tantas outras por Martin e Crowley, 1995 [15], é calcular a Soma das Diferenças Quadradas (sum of squared differences ou SSD) entre os pixels correspondentes das imagens. Quanto menor o valor obtido pelo cálculo do SSD, menor a diferença entre as duas imagens comparadas. Se as imagens forem idênticas, o valor é zero. Assim sendo, o SSD pode ser interpretado como sendo a distância entre dois vetores em um espaço de dimensão w x h. Uma das desvantagens deste método é a sua sensibilidade em relação as mudanças nas condições de iluminação. Um exemplo para ilustrar essa deficiência é o seguinte: se a mão se move para uma região escura da cena, a diferença em tons de cinza em relação aos pixels correspondentes na imagem anterior será grande para todos os pixels, e o resultado do cálculo do SSD também será grande, resultando em uma pequena correlação entre as imagens. Assim como no método de segmentação por cores, para solucionar esse problema, o ideal é normalizar as imagens amostradas pelas suas luminosidades. Essa correlação cruzada normalizada é descrita em Lewis, 1995 [37].

De acordo com Hardenberg, 2001 [23], a correlação é uma técnica muito custosa, visto que ela se baseia em procurar uma imagem (padrão) dentro de todas as imagens capturadas. Uma maneira de se melhorar é definindo uma janela de busca ao redor da última posição conhecida do objeto. Como o tamanho da janela de busca influi diretamente na performance do sistema, o tamanho máximo dessa janela depende do número máximo de imagens que o sistema precisa processar por segundo. O tamanho dessa janela de busca deve ser configurado de tal maneira que a robustez alcançada seja razoável,

visto que, quanto maior o tamanho da janela, maior a robustez e, como consequência, maior o tempo de processamento.

Wang et al, 2003 [78], afirma que o principal problema da correlação é a perda do rastreamento quando o objeto de interesse realiza movimentos muito rápidos, o que proporciona duas situações: ou o objeto de interesse sai fora da janela de busca, ou a sua imagem fica irreconhecível (borrada). Uma outra dificuldade é a classificação incorreta de objetos parecidos com os objetos de interesse. Por exemplo, estamos rastreando as mãos, e a fonte de luz está em uma posição em que a sombra da mão aparece nitidamente na imagem. A sombra provavelmente será classificada como mão.

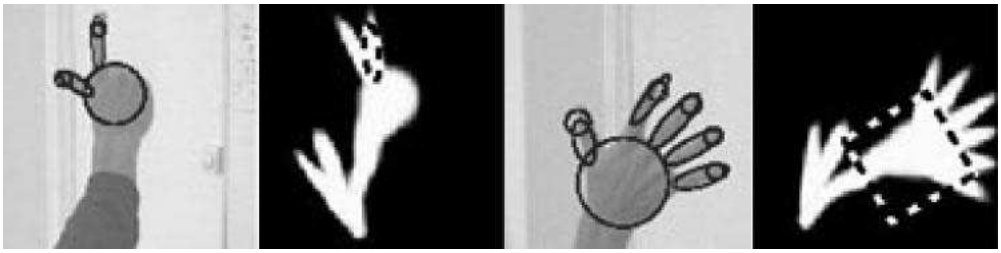

Figura 3.6: Aplicação Drawboard: um modelo é utilizado para achar a palma da mão, os dedos e as pontas dos dedos. Os estados da mão significam os comandos para o computador, como por exemplo, ferramenta de seleção ou de zoom (Laptev e Lindeberg, 2000 [36])

Em Laptev e Lindeberg, 2000 [36], foi desenvolvido um algoritmo que detecta confiavelmente a posição, orientação e escala da mão, assim como a configuração dos dedos, a uma tava de 10 hertz. A metodologia utilizada é a seguinte:

1. É criado uma hipótese sobre a posição e o estado da mão, baseado em um conjunto de treinamento, e também na distribuição de probabilidade das possiveis posições da mão na imagem;

2. Usa-se essa hípótese para construir um modelo de mão, que consiste em funções Gaussianas de duas dimensões;

3. Calcula-se a correlação entre o modelo de mão e os dados da imagem.

Esses três passos são repetidos com várias hipóteses diferentes. A hipótese que tiver menor valor de correlação, ou seja, a hipótese que for mais parecida com a imagem, é selecionada como a nova posição da mão. Esse algoritmo foi utilizado para construir uma aplicação de desenho chamada *DrawBoard* (Figura 3.6).

## 3.5 Rastreamento Baseado na Diferença entre Imagens

De acordo com Malm e Heyden, 2000 [41], a técnica de diferença entre imagens é baseada no sistema visual humano, o qual detecta o movimento de objetos primeiramente pela diferença na intensidade entre os mesmos e o fundo, não utilizando, a princípio, as variações cromáticas. Assim sendo, a técnica trabalha apenas com os valores dos pixels em tons de cinza, descartando as informações de cores. O seu algoritmo, no modo puro, é apenas a subtração de uma imagem (matriz) pela outra. Onde houver diferença entre as mesmas posições nas duas imagens, ou seja, onde a subtração retornar um valor diferente de zero, significa que naquela posição houve movimento. As desvantagens notadas facilmente neste técnica são: ela funciona apenas se existir contraste entre o objeto de interesse e o fundo da imagem, e a diferença é obtida tanto na nova posição do objeto como na antiga.

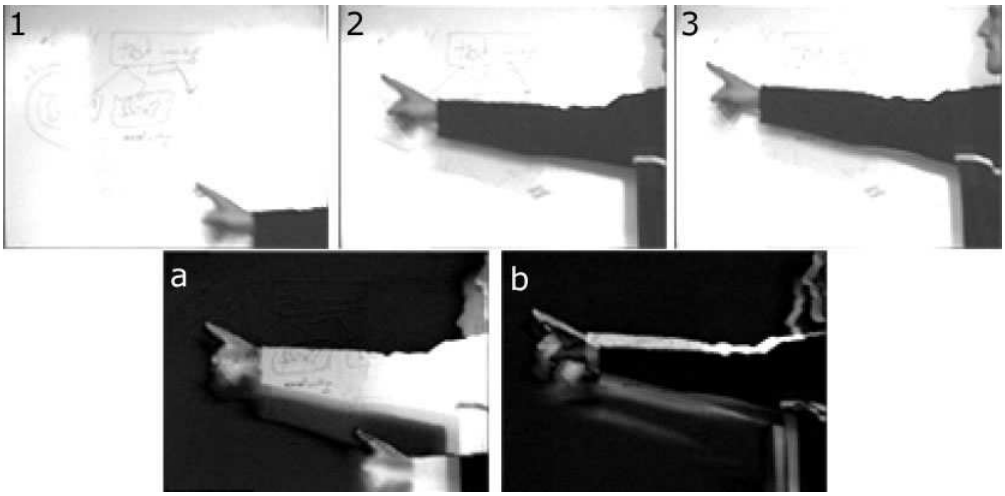


Figura 3.7: Diferença entre imagens: (a) Resultado de movimento grande entre a imagem 1 e a imagem 2 (b) Resultado de pequeno movimento entre a imagem 2 e a imagem 3. Ilustração reproduzida de Hardenberg, 2001 [23].

Um fator importante que deve ser lembrado é que pequenas variações na iluminação, vibrações induzidas (provocadas pelo vento calmo em uma cortina por exemplo), limitações eletrônicas do dispositivo de captura de imagens utilizado, interferências elétricas no circuito de vídeo, entre outros, acabam provocando pequenas diferenças de valores entre duas imagens consecutivas, imperceptíveis a olho nú, mas presentes. Isso atrapalha muito a técnica simples de diferença entre imagens, visto que, mesmo que a diferença entre os

valores seja realmente pequena, ela é diferente de zero, proporcionando a falsa classificação do pixel, como se naquela posição tivesse havido um movimento. E é aí que entra o limiar, que deve ser grande o suficiente para remover esses ruídos. Em experimentos realizados por Hardenberg, 2001 [23], os melhores resultados obtidos para diversas condições de iluminação utilizaram o limiar de valor 20 por cento, ou seja, a diferença só existe realmente se a diferença entre os dois pixels for superior a 20 por cento da maior diferença na escala de cinza. Como a maior diferença em tons de cinza que pode existir é de 255, assume-se que o valor ideal para o limiar é aproximadamente 51.

Em aplicações em que o fundo seja sempre mais claro que o objeto de interesse, como por exemplo, as mãos em frente a um quadro branco ou a uma parede, é possível melhorar um pouco o algoritmo de diferença entre imagens, não considerando os pixels com valor de cinza diferentes da imagem anterior, mas sim os pixels que são mais escuros que na imagem anterior, ou seja, os pixels do objeto de interesse. Assim, o resultado mostra apenas a nova posição do objeto, não aparecendo mais a posição anterior do mesmo. Ao contrário de outras técnicas já explanadas, esta funciona muito bem para grandes movimentos, porém não apresenta bons resultados para movimentos lentos. Uma possibilidade para continuar trabalhando com diferença entre imagens e evitar esse problema é não calcular a diferença entre imagens consecutivas, mas sim a diferença entra a imagem atual e a imagem de referência do fundo (imagem do fundo sem a presença do objeto de interesse).

A utilização do fundo como imagem de referência funciona bem apenas para aplicações que rodem por períodos bem curtos de tempo, visto que, na vida real, é muito difícil de se conseguir manter um fundo de cena estável por longos períodos de tempo: objetos são colocados e tirados das mesas, portas abrem e fecham, as condições de iluminação variam, a câmera muda de posição, entre outros. Uma solução para esse problema, proposta em Stafford-Fraser, 1996 [67], é atualizar a imagem de referência, durante o processamento, a cada ciclo de tempo pré-determinado. Quanto maior a frequência com que a imagem de referência é atualizada, maior a robustez. Por outro lado, se o objeto de interesse permanecer em inércia, ele acaba sendo adicionado à imagem de referência. Por esse motivo, o ideal é que a imagem de referência seja atualizada com um intervalo de tempo médio, como por exemplo, 1 minuto, visto que é muito difícil que as mãos fiquem paradas durante 1 minuto em uma interação homem-máquina. Uma outra alternativa é atualizar rapidamente, mas utilizar-se das informações do rastreamento para impedir que o algoritmo adicione os objetos de interesse ao fundo da imagem, mesmo que a mão não se movimente por longos períodos de tempo.

A técnica de diferença entre imagens, mesmo com todos os incrementos citados, é a técnica mais rápida, com apenas uma linha de código por

pixel, possibilitando, por esse motivo, o processamento de imagens de alta resolução em tempo real. As vantagens são que esta técnica não depende de imagens para amostra pré-selecionadas, e mudanças suaves na iluminação não interferem na segmentação.

## 3.6 Rastreamento Baseado em Contornos Ativos

O rastreamento baseado em contornos ativos, ou cobras (*snakes*), funciona extraindo diretamente o formato dos objetos (Baumberg e Hogg, 1994 [4]). A idéia é ter uma representação do contorno do objeto e manter-se atualizando esse contorno dinamicamente através do tempo (Bertalmío et al, 2000 [6]).

Essa técnica tem sido intensivamente estudada a alguns anos. Por exemplo, em Isard e Blake, 1996 [29], é adotada uma equação diferencial para descrever o movimento complexo do modelo. Já em Paragios e Deriche, 2000 [48], é apresentado uma estrutura para detecção e rastreamento de múltiplos objetos em movimento, baseando-se nos contornos ativos. Além desses, Peterfreund, 1999 [52], explorou um novo modelo de contornos ativos, baseado nos filtros de Kalman, para o rastreamento de objetos não-rígidos em movimento, como por exemplo, as mãos.

De acordo com Wang et al, 2003 [78], ao contrário do rastreamento baseado nas regiões, a vantagem de se ter um rastreamento baseado em contornos ativos é a redução na complexidade computacional. A desvantagem, entretanto, é que esta técnica requer um bom ajuste inicial. Se, de alguma forma, fosse possível inicializar um contorno separado para cada objeto em movimento, então todos seriam rastreados tranquilamente, mesmo na presença de oclusão parcial. A inicialização é muito difícil, especialmente para objetos altamente articulados, como as mãso.

## 3.7 Rastreamento Baseado em Características da Imagem

Abandonando a idéia de rastrear os objetos inteiros, esse método de rastreamento utiliza sub-características, tais como notáveis pontos ou linhas no objeto, para realizar a tarefa de rastreamento. Wang et al, 2003 [78] cita que o seu ponto forte é que, mesmo quando existe oclusão parcial, algumas das sub-características do objeto podem permanecer visíveis. O rastreamento baseado em características inclui a extração e a combinação de características.

Características de baixo nível, como por exemplo, pontos, são mais fáceis de serem extraídos. É relativamente mais difícil rastrear características de mais alto nível, como linhas. Então, nesta técnica também é importante o balanceamento entre a complexidade das características (velocidade de processamento) e a eficiência do rastreamento.

Um bom exemplo de rastramento baseado em pontos está em Polana e Nelson, 1994 [56], no qual, um retângulo virtual é colocado cercando a pessoa, e a centróide deste retângulo é selecionada como o ponto para o rastreamento. Mesmo em caso de oclusão entre dois objetos rastreados, se a velocidade das centróides pode ser distinguida, o rastreamento é realizado com sucesso.

Em Segen e Pingalli, 1996 [65], o rastreamento utiliza, como característica, os pontos dos cantos das silhuetas em movimento. Esses pontos são marcados usando uma medida de distância baseada nas posições e curvaturas dos mesmos

O rastreamento através de pontos e linhas baseado nos filtros de Kalman têm sido muito divulgado na área de visão computacional (Rosales e Sclaroff, 1998 [63]; Nguyen et al, 2001 [46]). Em um trabalho recente de Jang e Choi, 2000 [33], um molde ativo com características estruturais de cada região do objeto foi construído dinamicamente, usando como base as informações de formato, textura, cor e coordenadas das bordas da região.

## 3.8 Rastreamento Baseado em Momentos da Imagem

Em Starner e Pentland, 1995 [69], foram demonstrados resultados interessantes com técnicas relativamente simples: detecção de cores, crescimento de regiões e cálculo de momentos da imagem foram suficientes para construir um sistema que reconhece confiavelmente 40 palavras da língua Americana de Sinais. Para achar cada mão, inicialmente, o algoritmo percorre a imagem até achar um pixel de cor-da-pele. Dado esse pixel como semente, utiliza-se técnica de crescimento de regiões, que vai crescendo através dos 8 vizinhos mais próximos, verificando se os mesmos são de cor-da-pele. Cada pixel marcado fica sendo considerado como parte da mão. Feito isso, executa-se uma dilatação morfológica na imagem resultante, para prevenir falsos cortes na imagem, provocados por sombras, etc. A centróide é calculada e guardada como semente para a próxima imagem.

Outro método, apresentado em Lin et al, 2002 [39], apresenta como necessidades os rastreamentos local e global, para adquirir as informações completas da mão. O primeiro passo é detectar a orientação da mão. Uma boa

solução para isso é se utilizando de alguns momentos da imagem, para localizar a posição da palma da mão, que é onde normalmente se encontra o centro de massa da mão. Seguindo o contorno da palma da mão, se localiza o pulso, e assim é detectado a orientação da mão. A orientação da mão determina a orientação de todos os outros padrões a serem usados na estimação da localização e posição dos dedos.

## 3.9 Rastreamento Baseado no Fluxo Ótico

O fluxo ótico é calculado baseado no método de gradiente generalizado usando múltiplos filtros espaciais (Chen et al, 1993 [10]), assumindo que a região da mão na imagem possui um vetor de fluxos uniforme. A mão é rastreada atualizando-se uma janela retangular que a envolve.

Segundo Tsutsui et al, 2001 [72], para realizar o rastreamento baseado no fluxo ótico, considera-se que o primeiro objeto a se mover na imagem é a mão. Depois disso, é marcada uma janela na qual exista um número suficiente de vetores de fluxo, ou seja, uma janela onde haja movimento. Essa janela é chamada de janela de rastreamento. O rastreamento procede da seguinte forma (Figura 3.8):

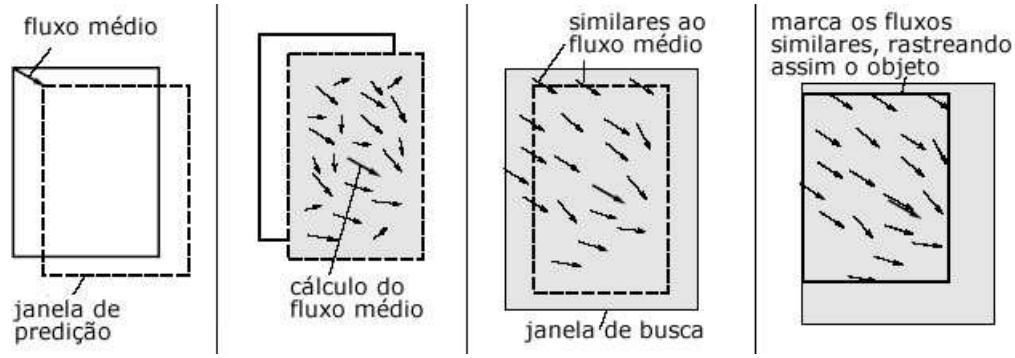


Figura 3.8: Rastreamento baseado no fluxo ótico.

1. A janela de rastreamento da imagem anterior é deslocada de acordo com o fluxo médio daquela imagem. Essa janela deslocada é chamada de janela de predição. Na imagem inicial, a janela de predição está na mesma posição da janela de rastreamento;

2. O fluxo médio é calculado na janela de predição;

3. Busca-se, na janela de predição e nas vizinhanças da mesma, os pixels com vetores de fluxo similares ao fluxo médio. A região do objeto é gerada com o conjunto desses pixels.

4. A janela de rastreamento é ajustada para envolver a região do objeto e, assim, a mão está rastreada.

# Capítulo 4

# Implementação do Rastreamento das Mãos no Conversor LIBRAS-Texto

Depois de ter estudado diversas técnicas, o conhecimento adquirido foi suficiente para adicionar ao conversor LIBRAS-texto este novo recurso: o rastreamento das mãos, ampliando assim a sua aplicabilidade, e facilitando o seu uso. Na primeira seção explicaremos um pouco do que é a Língua Brasileira de Sinais (LIBRAS). Já na segunda seção falaremos sobre o conversor LIBRAS-Texto, implementado em Pistori e Neto, 2003 [54]. Descreveremos as ferramentas utilizadas para o desenvolvimento do mesmo, assim como o seu funcionamento. Por último escreveremos a respeito da implementação do módulo de rastreamento no conversor.

## 4.1 A Língua Brasileira de Sinais - LIBRAS

A língua de brasileiras de sinais, LIBRAS, é a língua de sinais oficialmente reconhecida como meio legal de comunicação no Brasil. A comunicação em LIBRAS inclui movimentos de cabeça, tronco, braços e outras partes do corpo. Entretando, a forma da mão, sua posição em relação ao corpo do interlocutor e o movimento realizado por ela, são muitas vezes os blocos elementares para a construção de sentenças mais complexas. A língua LIBRAS inclui um conjunto de 46 formas básicas para as mãos, que são também chamadas configurações. Este conjunto inclui 19 símbolos alfabéticos executados através de posturas e 6 símbolos alfabéticos, correspondentes as letras h, j, k, x, y e z, representados através de gestos, conforme Pistori e Neto, 2003 [54]. A figura 4.1 representa alguns dos símbolos do alfabeto LIBRAS.

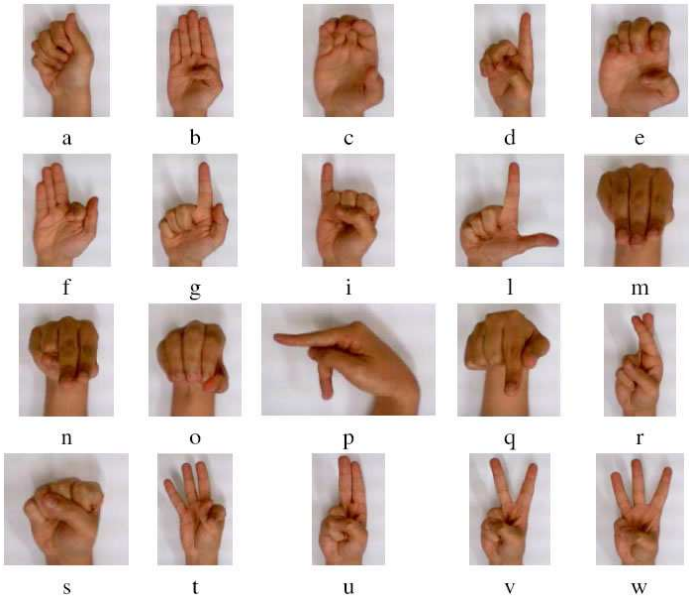


Figura 4.1: Imagens reproduzidas de Pistori e Neto, 2003 [54], demonstrando alguns símbolos do alfabeto LIBRAS.

## 4.2 O Conversor LIBRAS-Texto

Em Pistori e Neto, 2003 [54] foi desenvolvido um sistema que atua como um conversor LIBRAS-Texto, em que, em síntese, o usuário realiza o gesto de uma das letras do alfabeto LIBRAS, a câmera captura as imagens, através de técnicas de visão computacional o sinal é processado e, através de uma árvore de decisão adaptativa, transforma o gesto capturado em texto no computador. Este protótipo já apresenta bons resultados reconhecendo os padrões (gestos), entretanto é necessário que a mão esteja em uma posição pré-fixada, e o fundo necessita ser uniforme.

O protótipo foi implementado em Java, visando a portabilidade, e em seu desenvolvimento foram utilizadas as seguintes ferramentas:

1. Processamento Digital de Imagens - ImageJ [1]: Foi utilizado o pacote ImageJ, uma versão multiplataforma, ainda em desenvolvimento, do pacote NIH Image, para Macintosh. Além de ser um software livre com código aberto, este pacote tem como principal característica a disponibilidade de diversos algoritmos para manipular os mais variados formatos de imagens, melhorar a qualidade das imagens, realizar a detecção de bordas, operações morfológicas e diversos tipos de contas

[1] http://rsb.info.nih.gov/ij/

relacionadas ao processamento de imagens, como o cálculo das áreas, médias, centróides, etc. É possível adicionar ainda mais recursos, além de todos os que já vêm embutidos no pacote, através de plugins escritos em Java. Um outro fator que ajuda muito é a existência de um grande número de programadores trabalhando no seu desenvolvimento, o que possibilita a criação de novos plugins.

2. Aprendizagem de Máquina - WEKA [2]: O segundo pacote utilizado no desenvolvimento do conversor foi o WEKA (Waikato Environment for Knowledge Analysis), escrito em Java e com programas-fonte abertos. O WEKA é um ambiente bastante utilizado em pesquisas na área de aprendizagem de máquina, pois oferece diversos componentes que facilitam a implementaçãao de classificadores e agrupadores (clustering tools). É possível com este pacote obter facilmente resultados estatísticos comparativos da execução simultânea de diversos programas de aprendizagem em domínios variados, tais como o reconhecimento de caracteres, o reconhecimento de imagens e o diagnóstico médico, conforme escrito em Pistori e Neto, 2003 [54].

3. Tratamento de Dispositivos de Captura de Imagens - JMF [3]: E por último, um pacote para captura de sinais temporais em tempo real, o Java Media Framework. Este pacote permite que programas em Java possam acessar, por exemplo, imagens capturadas através de uma webcam acoplada a um computador pessoal. Segundo Pistori e Neto, 2003 [54], o software abstrai detalhes das arquiteturas e interfaces de diferentes dispositivos de captura de imagem, facilitando assim a portabilidade dos aplicativos.

O protótipo trabalha da seguinte maneira: primeiramente, ocorre a fase de treinamento, quando o usuário deve executar os sinais manuais, com uma das mãos, e utilizar a outra para digitar a letra correspondente. Diversas imagens da mão são capturadas, até que uma tecla qualquer seja pressionada. Este procedimento é repetido, diversas vezes, para todas as letras desejadas: normalmente, quanto mais exemplos forem colhidos na fase de treinamento, maior será a precisão obtida na fase de reconhecimento. Quando uma tecla especial, reservada, for pressionada, o sistema induz a árvore de decisão adaptativa inicial, e o usuário pode então iniciar a edição do texto, quando apenas os sinais manuais passam a ser necessários. Cada sinal manual deve ser mantido por cerca de meio segundo para que o sistema tenha tempo de

[2] http://www.cs.waikato.ac.nz/ ml/weka/

[3] http://java.sun.com/products/java-media/jmf/

reconhecer três quadros consecutivos com a mesma letra (quando então a letra é efetivamente digitada). O módulo de aprendizagem pode ser acionado a qualquer momento (por exemplo, quando o sistema comete um erro de classificação), bastando para isto digitar a letra correspondente ao símbolo que está sendo mostrado para a câmera. Quando isto ocorre, a árvore de decisão adaptativa incorpora os novos exemplos de treinamento, conforme descrito em Pistori e Neto, 2003 [54].

## 4.3 O Módulo de Rastreamento das Mãos no Conversor LIBRAS-Texto

Tendo como base todo o estudo feito sobre as técnicas de rastreamento, implementamos no conversor LIBRAS-texto o rastreamento das mãos, ampliando a sua aplicabilidade. Utilizaremos, no conversor, a técnica aplicada em Starner e Pentland, 1995 [69], por ser a técnica mais fácil de ser adaptada ao conversor dentre as estudadas, e por apresentar bons resultados.

Essa primeira imagem deve conter apenas a mão como objeto com cor-da-pele, e ela deve estar aberta, com a palma e os dedos bem aparentes, conforme figura 4.2a. Isso deve ser feito para o cálculo dos limites utilizados nas heurísticas do filtro. Com a imagem da mão aberta, têm-se aproximadamente o número máximo de pixels possíveis. Levando-se em conta apenas a palma da mão, têm-se aproximadamente o número mínimo de pixels possíveis (similar ao número de pixels presentes quando a mão aparece fechada).

A técnica utiliza, como um primeiro passo, a segmentação por cor-da-pele, cujo algoritmo já se encontra implementado no conversor desde a sua primeira versão. De acordo com Pistori, autor do conversor, o rastreamento por cor-da-pele já funciona no protótipo, com o sistema desenhando um retângulo que acompanha o movimento das mãos. Porém, trata-se de uma solução imperfeita, visto que, no caso de aparecimento de outro pedaço de pele na imagem (como por exemplo, a face), o rastreamento das mãos falha, rastreando qualquer objeto que tenha cor-da-pele. Para iniciar o protótipo, deve-se primeiro marcar uma região de cor-da-pele na imagem. Isso é requisito básico para o processo de segmentação por cores, criando-se um modelo a cada utilização do sistema, visto que as condições de iluminação podem estar diferentes das registradas na utilização anterior.

Para resolver esse problema, utiliza-se a segunda parte da solução proposta. Uma operação morfológica de dilatação é executada na imagem segmentada, para prevenir falsos cortes, provocados por sombras, por exemplo. Depois, executa-se o seguinte algoritmo:

1. A imagem é percorrida até que o algoritmo encontre um pixel de cor-da-pele ainda não visitado, criando aí uma nova candidata a região da mão;

2. Dado esse pixel como semente, utiliza-se a técnica de crescimento de regiões, que vai se expandindo recursivamente através dos 4 pixels adjacentes, visitando-os caso eles sejam de cor-da-pele. Cada pixel marcado fica sendo considerado como parte da região criada no passo 1, e as suas coordenadas vão sendo guardadas para o cálculo dos momentos da imagem. Retorna ao passo 1 até marcar todas as regiões;

3. Terminada a recursão, passa-se um filtro que elimina as regiões candidatas baseado na quantidade de pixels da região, que deve ser maior que o número mínimo de pixels estimado para a mão e menor que o número máximo de pixels;

4. Com as candidatas restantes, verifica-se qual a região com centro de massa mais próximo da região da mão na imagem anterior e, finalmente, a mão está rastreada. O centro de massa da mão rastreada é utilizado como semente para a técnica de crescimento de regiões da próxima imagem. É bem provável que este pixel faça parte da mão na próxima imagem, a não ser que haja oclusão, ou a mão tenha se movido para outra região muito rapidamente. Nesses casos, inicia-se o rastreamento novamente, a partir do primeiro passo.

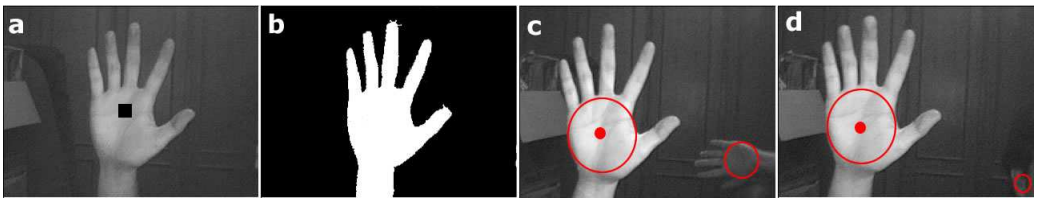


Figura 4.2: Imagens do processo de rastreamento implementado: (a) Primeira imagem capturada, com a mão aberta. O quadrado preto marca os pixels selecionados como cor-da-pele. (b) Imagem segmentada pela cor-da-pele. Nessa etapa já são extraídos os limites mínimo e máximo de quantidade de pixels. (c) As duas mãos são classificadas corretamente, pois a segunda mão não é menor do que o número mínimo de pixels estipulados. A mão correta é selecionada devido a sua maior proximidade com o centro de massa encontrado no rastreamento da imagem anterior. (d) Um pedaço de orelha é marcado como região, mas é desclassificado devido à regra do tamanho mínimo de pixels.

# Capítulo 5

# Conclusões

É de comum saber que, como a complexidade das aplicações vêm crescendo ao longo do tempo, tem crescido também a necessidade por dispositivos de interação diferentes dos convencionais (teclado, *mouse* e *joystick*). Uma alternativa interessante é utilizar-se dos meios naturais de comunicação e interação humana para o meio de interação homem-computador. Através do rastreamento das mãos, por exemplo, é possível realizar o reconhecimento de gestos, ou até mesmo controlar os canais e o volume de uma televisão. Com um número muito grande de pessoas trabalhando em soluções de rastreamento baseadas em visão, existe uma quantidade muito grande de técnicas já desenvolvidas.

Algumas técnicas fazem uso de luvas coloridas ou luvas com sensores eletro-mecânicos, outras utilizam hardware específico, sempre visando obter um melhor resultado. Essas são alternativas invasivas e caras. As técnicas de rastreamento baseadas apenas na visão computacional estão sendo as mais utilizadas, já que não requerem nenhum dispositivo adicional (luvas, etc.), e podem ser implementadas com imagens capturadas de uma simples webcam, tornando o seu baixo custo um grande atrativo.

Existe um número muito grande de pessoas trabalhando com soluções de rastreamento baseadas em visão, com uma quantidade vasta de técnicas diferentes já desenvolvidas, como por exemplo: soluções baseadas em filtros de Kalman (Vaillant e Darmos, 1995 [76]), detecção da cor da pele (Hongo et al, 2000 [26]), ou mesmo utilizando modelos de mãos 3D (Heap e Hogg, 1996 [25]). A pesquisa feita neste trabalho se concentrou no estudo de algumas técnicas não-invasivas de rastreamento das mãos bem distintas, que fazem uso apenas da visão computacional para resolver o problema. Os requisitos para a pesquisa das técnicas foram: não usar nenhum dispositivo adicional, visando aproximar a interação homem-máquina da interação natural entre homens e baratear o custo, e que as técnicas apresentassem bons

resultados em imagens com fundo heterogêneo. Não foram pesquisados apenas os métodos de rastreamento das mãos, mas também técnicas voltadas para outras aplicações da visão computacional, como por exemplo técnicas de rastreamento de objetos em geral, visando ampliar os conhecimentos gerais de visão computacional.

Um bom exemplo de utilidade para técnicas de rastreamento das mãos é o sistema desenvolvido em Pistori e Neto, 2003 [54], que atua como um conversor LIBRAS-texto, em que o usuário realiza o gesto de uma das letras do alfabeto LIBRAS, a câmera captura o gesto e o transforma em texto no computador. Entretanto é necessário que a mão esteja em uma posição préfixada, e o fundo necessita ser uniforme. Para aumentar a sua aplicabilidade e facilitar o seu uso, achou-se interessante sanar essas limitações do sistema, e por isso decidiu-se realizar este estudo comparativo entre algumas ténicas de rastreamento das mãos, para que com o conhecimento adquirido, fosse possível adicionar à este conversor este novo recurso: o rastreamento das mãos.

Os experimentos foram feitos com o auxílio do pacote ImageJ, uma versão multiplataforma, ainda em desenvolvimento, do pacote NIH Image, para Macintosh, no qual é possível a inclusão de recursos através de módulos escritos em Java. Assim, foi possível efetuar alguns testes antes de integrar o rastreamento ao conversor. Ou seja, só foi feito a integração de fato depois de selecionada a técnica com base nos testes realizados com o ImageJ.

Foi desenvolvido no conversor a melhor opção de rastreamento das mãos selecionada com base nos estudos feitos, e com base no atual estado do conversor. O desenvolvimento foi feito utilizando a linguagem Java, para não ser preciso reescrever o conversor, e também devido à orientação à objetos e à portabilidade, proporcionadas por essa linguagem.

Apesar de ainda necessitar de um estágio de inicialização, a utilização do conversor LIBRAS-texto se tornou muito mais fácil. Um próximo passo seria adicionar a este conversor o reconhecimento de sinais de outras letras, e de expressões, para as quais o rastreamento é primordial, visto que não se tratam apenas de posturas estáticas da mão, mas sim de gestos completos. Além disso, o rastreamento de múltiplos objetos também é um passo importante a ser desenvolvido, visando o reconhecimento de sinais de mais de uma pessoa ao mesmo tempo, ou mesmo dos sinais que fazem uso das duas mãos para expressar a informação desejada.

# Referências Bibliográficas


[1] R. Adams e L. Bischof. Seeded region growing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(6):641–647, 1994. ISSN 0162-8828.

[2] S. Ahmad. A usable real-time 3d hand tracker. In *28th Asilomar Conference on Signals, Systems and Computers*, páginas 1257–1261. IEEE Computer Society Press, 1995.

[3] A. Azarbayejani, C. Wren, e A. Pentland. Real-time 3-d tracking of the human body, 1996.

[4] A. M. Baumberg e D. C. Hogg. An Efficient Method for Contour Tracking Using Active Shape Models. Relatório Técnico 94.11, April 1994.

[5] G. Bebis, F. Harris, A. Erol, e B. Yi. Development of a nationally competitive program in computer vision technologies for effective human-computer interaction in virtual environments. *Space Grant, EPSCoR Annual Meeting*, Nov 2002.

[6] M. Bertalmío, G. Sapiro, e G. Randall. Morphing active contours. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(7):733–737, 2000. ISSN 0162-8828.

[7] J. M. Bradshaw. An introduction to software agents. In Jeffrey M. Bradshaw, editor, *Software Agents*, páginas 3–46. AAAI Press / The MIT Press, 1997.

[8] L. W. Campbell e A. F. Bobick. Recognition of human body motion using phase space constraints. In *ICCV*, páginas 624–630. 1995.

[9] J. Cassell, T. W. Bickmore, M. Billinghurst, L. Campbell, K. Chang, H. H. Vilhjalmsson, e H. Yan. Embodiment in conversational interfaces: Rea. In *CHI*, páginas 520–527. 1999.

[10] H.J. Chen, Y. Shirai, e M. Asada. Detecting multiple rigid image motions from an optical flow field obtained with multi-scale filters. In *IEICE Trans. Inf. and Syst., Vol. E76-D, No. 10*, páginas 1253–1262. 1993.

[11] S. Chen, W. Lin, e C. Chen. Split-and-merge image segmentation based on localized feature analysis and statistical tests. *CVGIP: Graph. Models Image Process.*, 53(5):457–475, 1991. ISSN 1049-9652.

[12] T. Cheng e C. Huang. Color images segmentation using scale space filter and markov random fields. In *Intelligent Robots and Computer Vision X : Algorithms and Techniques*, volume 1607, páginas 358–368. 1991.

[13] R. Cipolla e N.J. Hollinghurst. Human-robot interface by pointing with uncalibrated stereo vision. In *Image and Vision Computing*, páginas 171–178. apr 1996.

[14] C. Colombo e A. Bimbo. Interaction through eyes. *Robotics and Autonomous Systems*, 19:359–368, 1997.

[15] J. Crowley e J. Martin. Experimental comparison of correlation techniques, 1995.

[16] J. L. Crowley e J. Coutaz. Vision for man machine interaction. In *EHCI*, páginas 28–45. 1995.

[17] J.L. Crowley, J. Coutaz, e F. Bérard. Perceptual user interfaces: things that see. *Commun. ACM*, 43(3):54–ff., 2000. ISSN 0001-0782.

[18] S. Fiori e F. Piazza. A comparison of three *pca* neural techniques. In *ESANN proceedings - European Symposium on Artificial Neural Networks*, páginas 275–280. D-Facto public, apr 1999. ISBN 2-600049-9-X.

[19] W. T. Freeman e Weissman C. D. Television control by hand gestures. In *International Workshop on Automatic Face and Gesture Recognition*. jun 1995.

[20] R. C. Gonzalez e R. E. Woods. *Digital image processing*. Addison-Wesley, 1992.

[21] V. Grassi Júnior. *Sistema de visão omnidirecional aplicado no controle de robôs móveis*. Tese de Mestrado, USP, may 2002.

[22] P. Hamette, P. Lukowicz, e G. Tröster. Fingermouse: A wearable hand tracking system. In *UNICOMP Conference*. 2002.

[23] C. Hardenberg. *Fingertracking and Handposture Recognition for Real-Time Human-Computer Interaction*. Tese de Mestrado, Université Joseph Fourier, Grenoble, France, July 2001.

[24] C. Hardenberg e F. Bérard. Bare-hand human-computer interaction. In *Proceedings of the 2001 workshop on Percetive user interfaces*, páginas 1–8. ACM Press, 2001.

[25] A.J. Heap e D.C. Hogg. Towards 3d hand tracking using a deformable model. In *2nd International Face and Gestural Recognition Conference*, páginas 140–145. oct 1996.

[26] H. Hongo, M. Ohya, M. Yasumoto, e K. Yamamoto. Face and hand gesture recognition for human-computer interaction. In *Proceedings of the International Conference on Pattern Recognition - ICPR00*. IEEE, 2000.

[27] C. Huang, W. Huang, e C. Lien. Sign language recognition using 3-d hopfield neural network. In *Proceedings of the 1995 International Conference on Image Processing (Vol.2)-Volume 2*, página 2611. IEEE Computer Society, 1995. ISBN 0-8186-7310-9.

[28] G. Iannizzotto, M. Villari, e L. Vita. Hand tracking for human-computer interaction with graylevel visualglove: Turning back to the simple way. In *Workshop on Perceptive User Interfaces*. ACM Digital Library, November 2001. ISBN 1-58113-448-7.

[29] M. Isard e A. Blake. Contour tracking by stochastic propagation of conditional density. In *ECCV (1)*, páginas 343–356. 1996.

[30] M. Isard e A. Blake. Condensation – conditional density propagation for visual tracking, 1998.

[31] M. Isard e J. MacCormick. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *6th European Conference on Computer Vision*, páginas 3–19. 2000.

[32] A. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.

[33] D. Jang e H. Choi. Active models for tracking moving objects. *Pattern Recognition*, 33(7):1135–1146, 2000.

[34] T. Kulessa e M. Hoch. Efficient color segmentation under varying illumination conditions, 1998.

[35] M. La Cascia, S. Sclaroff, e V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models. Relatório Técnico 1999-005, Dipartimento di Informatica e Sistemistica, Universitá di Pavia, 23, 1999.

[36] I. Laptev e T. Lindeberg. Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features. In *Proceedings of the Third International Conference on Scale-Space and Morphology in Computer Vision*, páginas 63–74. Springer-Verlag, 2001. ISBN 3-540-42317-6.

[37] J. P. Lewis. Fast normalized cross-correlation, 1995.

[38] L. Lieberman. Autonomous Interface Agents. In *Proceedings of the ACM Conference on Computers and Human Interface, CHI-97*. Atlanta, Georgia, 1997.

[39] E. Lin, A. Cassidy, D. Hook, A. Baliga, e T. Chen. Hand tracking using spatial gesture modeling and visual feedback for a virtual dj system. In *Fourth IEEE International Conference on Multimodal Interfaces (ICMI'02)*. 2002.

[40] S. Lu, D. Metaxas, D. Samaras, e J. Oliensis. Using multiple cues for hand tracking and model refinement. In *WMCV*. 2002.

[41] H. Malm e A. Heyden. Hand-eye calibration from image derivatives. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, páginas 493–507. Springer-Verlag, 2000. ISBN 3-540-67686-4.

[42] S. Marcel. Gestures for multi-modal interfaces: A review. IDIAP-RR 34, IDIAP, 2002.

[43] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, e H. Wechsler. Tracking groups of people. *Computer Vision and Image Understanding: CVIU*, 80(1):42–56, 2000.

[44] R. A. F. Mini e M. F. M. Campos. Visual tracking of objects using multiresolution. In *XII Brazilian Symposium on Computer Graphics and Image Processing*, página 153. oct 1999.

[45] Y. Nam e K. Wohn. Recognition of space-time handgestures using hidden markov model. In *ACM Symposium on Virtual Reality Software and Technology*, páginas 51–58. 1996.

[46] H. Nguyen, M. Worring, e R. Boomgaard. Occlusion robust adaptive template tracking. páginas 678–683. 2001.

[47] Y. Ohno, J. Miura, e Y. Shirai. Tracking players and estimation of the 3d position of a ball. In *ICPR*, páginas 145–148. 2000.

[48] N. Paragios e R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(3):266–280, 2000.

[49] V. Pavlovic, J. M. Rehg, T. Cham, e K. P. Murphy. A dynamic bayesian network approach to figure tracking using learned dynamic models. In *ICCV (1)*, páginas 94–101. 1999.

[50] V. I. Pavlovic, R. Sharma, e T. S. Huang. Gestural interface to a visual computing environment for molecular biologists. In *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, página 30. IEEE Computer Society, 1996. ISBN 0-8186-7713-9.

[51] A. Pentland. Perceptual user interfaces: perceptual intelligence. *Commun. ACM*, 43(3):35–44, 2000. ISSN 0001-0782.

[52] N. Peterfreund. Robust tracking of position and velocity with kalman snakes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(6):564–569, 1999. ISSN 0162-8828.

[53] H. Pistori e J. J. Neto. An experiment on handshape sign recognition using adaptive technology: Preliminary results. In *Lecture Notes in Artificial Intelligence - Vol. 3171 - Special issue - XVII Brazilian Symposium on Artificial Intelligence - SBIA04*. Springer-Verlag, oct 2004.

[54] H. Pistori e J.J. Neto. *Tecnologia Adaptativa em Engenharia de Computação: Estado da Arte e Aplicações*. Tese de Doutoramento, USP, São Paulo, Brasil, Dec 2003.

[55] H. Pistori, J.J. Neto, A.A. Castro, M.C. Pereira, e T.R.V. Santos. Sigus - plataforma de apoio ao desenvolvimento de sistemas para inclus ao digital de pessoas com necessidades especiais, oct 2004.

[56] R. Polana e R. Nelson. Low level recognition of human motion, 1994.

[57] M. Porta. Vision-based user interfaces: methods and applications. *International Journal of Human-Computer Studies*, 57(1):27–73(47), jul 2002.

[58] F. Quek. Toward a vision-based hand gesture interface. In *Proceedings of the conference on Virtual reality software and technology*, páginas 17–31. World Scientific Publishing Co., Inc., 1994. ISBN 981-02-1867-2.

[59] A. Ramamoorthya, N. Vaswania, S. Chaudhurya, e S. Banerjeeb. Recognition of dynamic hand gestures. In *Pattern Recognition 36*. Elsevier Science Ltd., 2003.

[60] J. M. Rehg. *Visual Analysis of High DOF Articulated Objects with Application to Hand Tracking*. Tese de Doutoramento, Dept of Computer Science, 1995.

[61] J.M. Rehg e T. Kanade. Digiteyes: vision-based human hand tracking. Relatório Técnico CMU TR CMU-CS-93-220, CMU Technical Report, 1993.

[62] J.M. Rehg e T. Kanade. Visual tracking of high dof articulated structures: an application to human hand tracking. In *3rd European Conference on Computer Vision*, páginas 35–46. Springer Verlag, may 1994.

[63] R. Rosales e S. Sclaroff. 3D trajectory recovery for tracking multiple objects and trajectory guided recognition of actions. Relatório Técnico 1998-019, 4, 1998.

[64] B. Schiele e A. Waibel. Gaze tracking based on face-color, 1995.

[65] J. Segen e S. G. Pingali. A camera-based system for tracking people in real time. In *Proceedings of the International Conference on Pattern Recognition (ICPR '96) Volume III-Volume 7276*, página 63. IEEE Computer Society, 1996. ISBN 0-8186-7282-X.

[66] H. Sidenbladh, M. J. Black, e D. J. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *ECCV (2)*, páginas 702–718. 2000.

[67] J.Q. Stafford-Fraser. *Video-Augmented Environments*. Tese de Doutoramento, Gonville Caius College, University of Cambridge, feb 1996.

[68] J.Q. Stafford-Fraser e P. Robinson. Brightboard: A video-augmented environment. In *HCI*, páginas 134–141. 1996.

[69] T. Starner e A. Pentland. Real-time american sign language recognition from video using hidden markov models. In *Proceedings of the International Symposium on Computer Vision*, página 265. IEEE Computer Society, 1995. ISBN 0-8186-7190-4.

[70] D. J. Sturman. *Whole-hand input*. Tese de Doutoramento, Massachusetts Institute of Technology, 1992.

[71] K. Toyama e G. Hager. Incremental focus of attention for robust visual tracking. In *Internation Journal of Computer Vision v.35*, páginas 45–53. 1999.

[72] H. Tsutsui, J. Miura, e Y. Shirai. Optical flow-based person tracking by multiple cameras. In *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, páginas 91–96. 2001.

[73] M. Turk. Moving from guis to puis. In *Proc. Symposium on Intelligent Information Media*. dec 1998.

[74] M. Turk e G. Robertson. Perceptual user interfaces. *Communications of the ACM*, 43(3):32–34, 2000.

[75] S. E Umbaugen. *Computer vision and image processing: a practical approach using CVIPtools*. Prentice Hall, 1998.

[76] R. Vaillant e D. Darmon. Vision based hand pose estimation. In *International Workshop on Automatic Face and Gesture Recognition*, páginas 356–361. jun 1995.

[77] A. Waibel e M. T. Vo. A multi-modal human-computer interface: Combination of gesture and speech recognition. In *Proceedings of Inter Conference on Human Factors in Computing Systems*. apr 1993.

[78] L. Wang, W. Hu, e T. Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, 2003.

[79] G. Welch e G. Bishop. An introduction to the kalman filter. Relatório técnico, 1995.

[80] C. R. Wren, A. Azarbayejani, T. Darrell, e A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.

[81] C. R. Wren e A. P. Pentland. Understanding purposeful human motion. In *Proceedings of the IEEE International Workshop on Modelling People*, página 19. IEEE Computer Society, 1999. ISBN 0-7695-0362-4.

[82] Y. Wu e T. Huang. An adaptive self-organizing color segmentation algorithm with application to robust realtime human hand localization, 2000.

[83] X. Zhu, J. Yang, e A. Waibel. Segmenting hands of arbitrary color, 2000.

[84] V. D. Zwaan e J. Santos-Victor. Real-time vision-based station keeping for underwater robots. In *IEEE Oceans 2001*. Honolulu, nov 2001.