Universidade Federal da Bahia
Programa de Pós-graduação em Mecatrônica

Detecção e rastreamento da mão utilizando dados de profundidade

Thalisson Nobre Santos

2015

Detecção e rastreamento da mão utilizando dados de profundidade

Thalisson Nobre Santos

Dissertação submetida
como requisito parcial para obtenção
do grau de Mestre em Mecatrônica.

Programa de Pós-Graduação em Mecatrônica
Universidade Federal da Bahia

Sob a supervisão do
Prof. Dr. Luciano Rebouças de Oliveira (Orientador)

S237 Santos, Thalisson Nobre.

Detecção e rastreamento da mão utilizando dados de profundidade / Thalisson Nobre Santos. – Salvador, 2016.

77 f. : il. color.

Orientador: Prof. Dr. Luciano Rebouças de Oliveira.

Dissertação (mestrado) – Universidade Federal da Bahia. Escola Politécnica, 2016.

1. Interação homem - Máquina. 2. Processamento de imagens. 3. Visão por computador . I. Oliveira, Luciano Rebouças de. II. Universidade Federal da Bahia. III. Título.

CDD.: 621.367

TERMO DE APROVAÇÃO

THALISSON NOBRE SANTOS

DETECÇÃO E RASTREAMENTO DA MÃO UTILIZANDO DADOS DE PROFUNDIDADE

Dissertação aprovada como requisito parcial para a obtenção do grau de Mestre em Mecatrônica, Universidade Federal da Bahia, pela seguinte banca examinadora:

Orientador:
Prof. Dr. Luciano Rebouças de Oliveira
UFBA

Membro Interno:
Prof. Dr. Maurício Pamplona Segundo
UFBA

Membro externo:
Profa. Dra. Michele Fúlvia Angelo
UEFS

Salvador, 16 de dezembro de 2015

"Tudo posso naquele que me fortalece."

A Bíblia sagrada (Filipenses 4:13)

Resumo

As interfaces naturais têm demonstrado uma grande importância na interação entre o homem e a máquina, viabilizando desde jogos eletrônicos até a reabilitação de pacientes submetidos a fisioterapia. O rastreamento da mão por câmeras permite implementar tais interfaces, explorando os gestos humanos para controlar algum sistema computadorizado sem a necessidade de contato físico. O método proposto neste trabalho visa detectar e rastrear as mãos utilizando dados de profundidade. Uma vez que tais dados não produzem quantidade suficiente de pontos de interesse (pontos chaves) para a detecção da mão, foi proposto um algoritmo denominado Volume da Normal para exceder a descrição das características presentes nestas imagens, sendo baseado no cálculo do volume do vetor normal de cada pixel atribuindo valores arbitrários para o tamanho deste vetor. O rastreamento da mão é baseado na análise de descritores locais da imagem de profundidade (processada pela Transformada da Distância Euclidiana) e de um conjunto de imagens da mão após aplicação do Volume da Normal, utilizando para isto o algoritmo Oriented FAST and Rotated BRIEF. Um procedimento para a criação de um modelo cinemático da mão foi proposto como estágio inicial para um possível rastreamento contínuo dos dedos numa pesquisa posterior. Ao final, a detecção da mão foi executada a uma velocidade de 7,9 quadros por segundo, alcançando uma taxa de detecção média para detecção de poses do conjunto de treinamento igual a 36,4% e 38,15% para poses variadas. Para detecção de gestos realizados a partir do conjunto de treinamento foi alcançada uma taxa média de 21,94%. Para cenários onde há presença de objetos semelhantes à mão, o detector apresentou uma taxa de precisão igual a 14,72% com um desvio padrão de 3,82%.

Sumário

Lista de Figuras

Lista de Tabelas

Abreviações

ACP	**A**nálise de **C**omponentes **P**rincipais
ADV-4	**A**cúmulo da **D**iferença da **V**izinhança 4
BRIEF	**B**inary **R**obust **I**ndependent **E**lementary **F**eatures
CAMShift	**C**ontinuously **A**daptive **M**ean-Shift
CAA	**C**onsenso entre **A**mostras **A**leatórias
DBN	**D**eep **B**elief **N**etworks
EBM	**E**lliptical **B**oundary **M**odel
FAST	**F**eatures from **A**ccelerated **S**egment **T**est
FK	**F**iltro de **K**alman
HGO	**H**istograma de **G**radientes **O**rientados
IHC	**I**nteração **H**omem **C**omputador
INU	**I**nterface **N**atural do **U**suário
KTSL	**K**inect-based **T**aiwanese **S**ign-Language **L**
KVP	**K** **V**izinhos mais **P**róximos
MVS	**M**áquina de **V**etores de **S**uporte
MS	**M**ean **S**hift
OpenCV	**O**pen **C**omputer **V**ision
OpenNI	**O**pen **N**atural **I**nteraction
ORB	**O**riented FAST **R**otated **B**RIEF
RGB	**R**ed **G**reen **B**lue
RGBD	**R**ed **G**reen **B**lue **D**epth
SIFT	**S**cale-**I**nvariant **F**eature **T**ransform
SURF	**S**peeded **U**p **R**obust **F**eatures
TDE	**T**ransformada da **D**istância **E**uclidiana
VNOR	**V**olume da **NOR**mal

Ao meu Deus, à minha família e a todos aqueles que de forma direta ou indiretamente contribuiram para a construção deste trabalho.

Capítulo 1

Introdução

Conteúdo

Desde sua criação, o computador tem estado cada vez mais presente na vida das pessoas para auxiliá-las nas mais variadas tarefas. Contudo, o seu manuseio sempre esteve condicionado ao uso de algum dispositivo físico, restringindo a sua forma de interação. À medida que a tecnologia evoluiu, surgiu a necessidade de obter uma interação mais confortável e eficaz entre o homem e a máquina. Desde a criação do mouse como principal meio de interação gráfica, foram explorados novos paradigmas para tornar a comunicação entre o homem e a máquina tão natural quanto a interação entre pessoas.

As diferentes formas de interação existentes na comunicação entre os seres humanos, como fala e gestos principalmente, foram aproveitadas para aprimorar a área de Interação Homem-Computador (IHC), constituindo uma sub-área desta, chamada Interface Natural do Usuário (INU)[1]. A INU explora os diversos tipos de comportamento inerente ao corpo humano (reconhecimento de voz, de gestos, de ações, rastreamento do olhar, entre outros) que podem ser aproveitados para o controle direto de diferentes dispositivos (computador, consoles de jogos, celulares) em diferentes áreas, desde entretenimento (WIGDOR; WIXON, 2011) até a área médica (OGIELA; HACHAJ, 2014).

Embora tenham surgidos diferentes dispositivos e tecnologias para proporcionar o uso das INUs, a Visão Computacional tem sido amplamente explorada e atualmente constitui um dos seus pilares (RAUTARAY; AGRAWAL, 2012). O surgimento do sensor

[1] NUI - Natural User Interface

Kinect, um dispositivo de baixo custo capaz de fornecer diferentes tipos de dados visuais: Vermelho, Verde, Azul e Profundidade (RGBD)[2], proporcionou o crescimento de pesquisas envolvendo reconhecimento de ações e gestos humanos.

Interfaces que são controladas apenas com os gestos da mão e que possuam uma ampla sintaxe de gestos requerem uma técnica precisa de rastreamento (BILLINGHURST; BUXTON, 2016). Em alguns casos, o uso de equipamentos rastreadores junto à mão facilita este procedimento, porém o seu custo é elevado e são necessários durante todo o processamento podendo gerar desconforto. Em contrapartida, aplicações em Visão Computacional têm sido amplamente desenvolvidas ao longo dos anos a fim de alcançar melhores resultados quanto ao uso das interfaces naturais.

1.1 Objetivos

Implementar uma solução para detectar a mão utilizando imagens com a informação de profundidade e que seja robusto o suficiente para viabilizar sua aplicação em interfaces naturais.

1.1.1 Objetivos específicos

- Criar suporte para posicionar o sensor Kinect verticalmente.
- Elaborar o rastreamento da mão.
- Extrair o modelo cinemático da mão.

1.2 Contribuições

A pesquisa desenvolvida apresenta duas principais contribuições: (i) um algoritmo para detectar bordas baseado no acúmulo da diferença entre pixels vizinhos; (ii) Uma maneira de extrair características em imagens de profundidade. Adicionalmente, foi demonstrado um procedimento para a criação do modelo cinemático da mão em um cenário específico.

Resultados alcançados neste trabalho foram publicados no Workshop de trabalhos em andamento (WTA)[3] da Conferência em Grafícos, Padrões e Imagens 2015 (SIBGRAPI)[4]

[2]RGBD - Red, Green, Blue e Depth
[3]WIP - Work in progress
[4]SIBGRAPI - Conference on Graphics, Patterns and Images

com o seguinte título: *Finger phalanx detection and tracking by contour analysis on RGB-D images* (SANTOS; OLIVEIRA, 2015).

1.3 Mapa dos Capítulos

O restante do documento é descrito como:

- O **Capítulo 2** descreve o estado-da-arte sobre detecção e rastreamento da mão em imagens (RGB e RGBD), bem como a análise de técnicas relacionadas ao nosso trabalho.
- O **Capítulo 3** aborda detalhadamente o procedimento utilizado em cada etapa do sistema proposto, sendo: (i) Pré-processamento; (ii) Extração das características da mão; (iii) Rastreamento da mão e o (iv) Modelo cinemático.
- O **Capítulo 4** analisa o desempenho do sistema proposto e suas etapas.
- O **Capítulo 5** conclui esta dissertação e apresenta trabalhos futuros.

Capítulo 2

Estado da Arte

Conteúdo

O processo de detecção da mão constitui o estágio fundamental para diversas abordagens envolvendo análise da mão em imagens. Entretanto, devido o alto grau de liberdade da mão, podem-se ter diferentes poses, tornando esta tarefa mais desafiadora. Nesta seção são descritas algumas abordagens que foram proposta na literatura como solução para este problema.

2.1 Detecção da mão

A mão tem uma grande flexibilidade em executar movimentos em relação a outros membros do corpo humano, visto a sua peculiar anatomia capaz de manipular coisas. Esta característica a faz assumir múltiplas formas sendo útil para as mais diversas tarefas, inclusive a realização de gestos. Estudos visando analisar a estrutura da mão e seu movimento utilizando dados visuais (imagens) para o uso nas mais variadas aplicações para IHC, necessitam cumprir um passo essencial e complexo que iniciará todo o procedimento: a localização inicial desta na imagem observada. Devido à habilidade em assumir múltiplas posturas, a mão é um dos elementos mais complexos de se identificar no âmbito de Visão Computacional, uma vez que sua estrutura possui 30 graus de liberdade (LEE; KUNII, 1995). Além disso, um conjunto de outros fatores torna o processo de detecção desafiador, seja o ambiente no qual a cena é capturada, as condições de iluminação presente no ambiente, a distância da mão em relação à câmera, entre outros.

Utilizando uma câmera RGBD como único meio de aquisição de imagens durante o processo de detecção da mão, podemos classificar os métodos em quatro principais categorias: segmentação por limiar, detecção por aparência, segmentação por corpo e outras abordagens.

2.1.1 Segmentação por limiar

O uso de um limiar para delimitar uma faixa de atuação na imagem tem sido o princípio básico de muitas abordagens, uma vez que o processo de detecção da mão se torna uma tarefa muito mais simples mediante a cooperação da pessoa posicionando as mãos na faixa de profundidade estabelecida. O trabalho proposto por Liang *et al.* (2012) utiliza os dados de profundidade para segmentar o plano de frente e prediz que a mão é o objeto mais próximo à câmera, e faz uma análise deste quanto à angulação, tamanho e morfologia para verificar se realmente é uma mão. Similarmente, Li (2012) segmenta o plano de frente com dois limiares fixos e posteriormente utiliza o algoritmo de *Agrupamento por K-Médias* para indicar grupo de pixels, nesta região, como possíveis mãos. A abordagem proposta por Giraldo *et al.* (2012) divide a imagem de profundidade em quatro regiões diferentes, utilizando uma dessas para indicar a presença da mão. Em geral, o processo de detecção da mão por limiar é trivial e tem como principal limitação à necessidade de posicionar a mão em uma determinada região da imagem de profundidade. Abordagens descritas a seguir exploram paradigmas mais complexos.

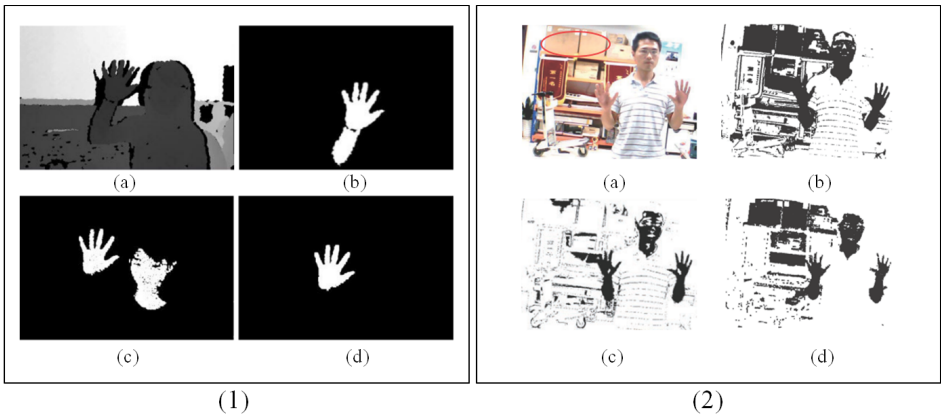


Figura 2.1: Tipos de segmentação utilizando informação de cor e de profundidade. 1) Segmentação baseada em dados de profundidade e cor proposto por Hongyong e Youling (2012): a) mapa de profundidade; b) mapa de profundidade após a limiarização; c) imagem RGB após o processo de detecção da pele; d) mão segmentada. 2) Segmentação baseada no método do EBM (XU *et al.*, 2011): a) imagem RGB; b) limiarização no espaço de cor RGB; c) aplicação do modelo Gaussiano; d) aplicação do método EBM. As regiões pretas indicam a cor da pele e o círculo vermelho indica as cores semelhantes à pele. Figuras extraídas de Hongyong e Youling (2012) e Xu *et al.* (2011), respectivamente.

2.1.2 Detecção por aparência

Utilizar dados da aparência da mão como fundamento para análise no processo de detecção pode ser considerada uma estratégia ousada devido aos vários fatores que influenciam diretamente no resultado final. Devido aos 30 graus de liberdade da mão, esta nem sempre apresenta uma forma definida: sombra, alterações no plano de fundo e variações na iluminação do ambiente são fatores que podem comprometer o resultado da detecção.

Combinando dados de profundidade (FIGURA 2.1-1a) e cor, Hongyong e Youling (2012) sugerem uma abordagem em que é estabelecida uma faixa de profundidade para descartar todos os objetos mais próximos e mais afastados do sensor Kinect (FIGURA 2.1-1b). Paralelamente, uma segmentação da cor da pele é realizada com o intuito de definir regiões que identifiquem a mão (FIGURA 2.1-1c); posteriormente, o resultado é usado para localização da mão (FIGURA 2.1-1d). O trabalho de Xu *et al.* (2011) apresenta um modelo robusto *Elliptical Boundary Model* (EBM) para detecção da cor da pele humana. De início é realizada uma limiarização no espaço de cor RGB (FIGURA 2.1-2b), em seguida é aplicado um modelo Gaussiano para detectar a pele (FIGURA 2.1-2c) e então é aplicado o EBM (FIGURA 2.1-2d). Ao final, a imagem de profundidade é utilizada para detectar o corpo humano e segmentar o plano de fundo da cena para distinguir objetos que apresentam cores semelhantes à pele humana (FIGURA 2.1-2a).

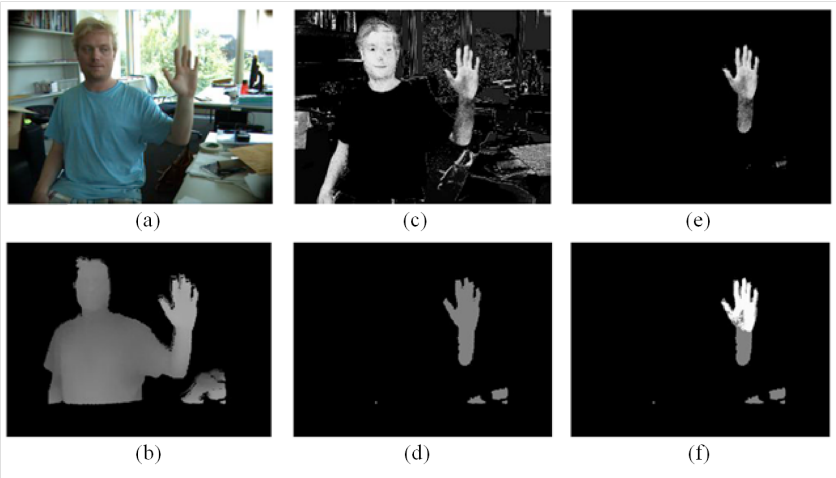


Figura 2.2: Abordagem proposta por Bergh e Gool (2011) para segmentação da região de interação das mãos. a) imagem RGB; b) imagem de profundidade; c) probabilidade da cor da pele após a abordagem híbrida (GMM e modelo de cor baseado em histograma); d) imagem de profundidade após limiarização; e) pixels da imagem *c* referente ao plano de frente; f) mão segmentada. Figura extraída de Bergh e Gool (2011).

O trabalho apresentado por Bergh e Gool (2011) propõe segmentar a cor da pele utilizando o Modelo de Mistura Gaussiana (MMG)[1] juntamente com outro modelo de cor de pele baseado em histograma (FIGURA 2.2-c). O valor médio referente à área do rosto, na imagem de profundidade (FIGURA 2.2-b), é utilizado como limiar para delimitar a região de interação com a mão e assim eliminar o plano de fundo da imagem (FIGURA 2.2-d,e,f). Embora tenha apresentado resultados acima de 95% em todos os cenários testados (considerando os dados de profundidade), o funcionamento do método está limitado à presença de algum rosto na imagem (FIGURA 2.2).

Os trabalhos apresentados nesta seção apresentam abordagens para a detecção da mão considerando, principalmente, informações referentes à aparência (cor). Embora funcionais, este tipo de abordagem pode apresentar falhas devido aos fatores inerentes ao uso de imagens RGB, seja a variação de iluminação ou a presença de objetos na cena com cor similar a pele humana.

2.1.3 Segmentação do corpo

O mapa de profundidade extraído de uma cena contém informações que proporcionam explorar possibilidades não alcançadas antes com a informação de intensidade de cor. O rastreamento do esqueleto do corpo humano é um destes benefícios e tem sido aplicado nos mais diversos contextos em Visão Computacional, desde jogos eletrônicos

[1] GMM - Gaussian Mixture Model.

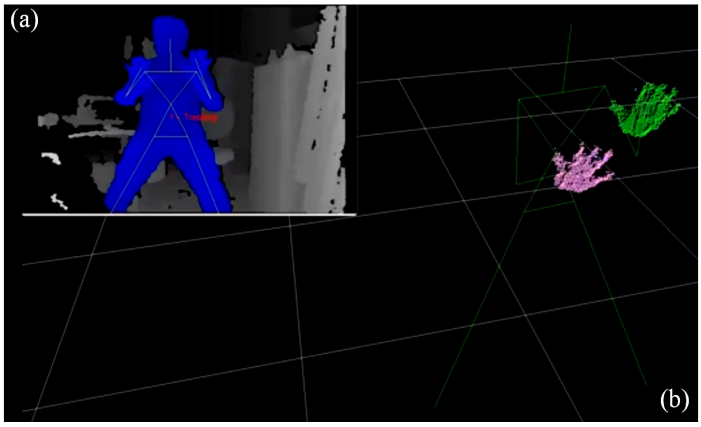


Figura 2.3: Abordagem proposta por Gallagher (2013) para detecção da região das mãos utilizando o esqueleto humano extraído a partir dos dados de profundidade. a) mapa de profundidade com o esqueleto humano detectado (azul); b) esqueleto humano (verde) com nuvem de pontos da mão (rosa e verde). Figura extraída de Gallagher (2013).

(BOZGEYIKLI *et al.*, 2013) até a tratamentos fisioterapeuticos por meio de reabilitação virtual (CAMPOS, 2013).

Uma estratégia interessante é utilizar o rastreamento do esqueleto para localizar o corpo da pessoa na cena e facilitar a detecção da mão. O trabalho de Phadtare *et al.* (2012) segmenta a região da mão em relação ao plano de fundo usando a posição da articulação do pulso como limiar, uma vez que as informações das articulações e membros do esqueleto são extraídas durante o seu rastreamento. Uma variação do rastreamento do esqueleto é apresentado por Gallagher (2013), onde uma nuvem de pontos da mão é extraída e fundida ao esqueleto (FIGURA 2.3). Assim, Chen *et al.* (2012) utilizam a nuvem de pontos da mão e fazem uma comparação com o mapa de profundidade para desconsiderar todos os pixels não pertencentes a região da mão com base em um limiar de dispersão estabelecido.

Outras abordagens utilizam a detecção do esqueleto com algumas variações, como a abordagem proposta por Kim e Rhee (2012), que além de utilizar o detector de esqueleto, sugere ainda uma abordagem utilizando diferenciação de quadros consecutivos do mapa de profundidade. Pisharady e Saerbeck (2013) usam os dados do esqueleto para recuperar as coordenadas x, y, z da mão direita, pescoço e ombros.

Apesar de ser uma escolha promissora, o rastreamento do esqueleto está sujeito a certas limitações, como a necessidade da presença deste posicionado ligeiramente à frente da câmera. Falhas durante a detecção devido à oclusão parcial de alguma parte do corpo e também o campo de visão da câmera deverá ser configurado de forma que seja possível capturar todo o corpo na cena. Em certas abordagens, não é possível utilizar a segmentação por corpo, como será mostrado na Seção 2.4.

Figura 2.4: Diferentes abordagens utilizas para detecção da mão. a) abordagem proposta por Cordella *et al.* (2012) utilizando marcadores (azul) sobre a mão; b) abordagem sugerida por Ren *et al.* (2011) para segmentar a mão com uma pulseira (preto) sobre o pulso; c) técnica para detectar a mão baseada em movimento proposto por Santos *et al.* (2011); d) trabalho sugerido por Minnen e Zafrulla (2011) identifica a mão como regiões extremas em relação algum centro de massa. Figuras extraídas de Cordella *et al.* (2012), Ren *et al.* (2011), Santos *et al.* (2011), Minnen e Zafrulla (2011), respectivamente.

2.1.4 Outras abordagens

Outros trabalhos que utilizam algum processamento envolvendo análise sobre os gestos da mão não tratam o processo de detecção como o cerne da pesquisa, mas, sendo este um pré-requisito, é possível utilizar artifícios que reduzem a complexidade do problema, como o uso de marcadores sobre a mão para facilitar a localização desta na imagem como visto na figura 2.4(a) (CORDELLA *et al.*, 2012). Ren *et al.* (2011) propõem uma forma de segmentar a região da mão utilizando uma pulseira preta sobre o pulso (FIGURA 2.4-b). Existem diferentes abordagens utilizando o movimento da mão previamente determinado para facilitar a detecção, tais como as apresentadas por Santos *et al.* (2011) (FIGURA 2.4-c) e Bergh e Gool (2011).

Utilizando o conceito de distância geodésica, Minnen e Zafrulla (2011) considera candidato potencial à mão qualquer região situada ao extremo do centro de massa do corpo humano (FIGURA 2.4-d).

2.2 Detecção dos dedos

O processamento de detecção dos dedos é um estágio complementar ao de detecção da mão, visto que com a localização destes últimos é possível obter maior proveito durante a utilização da mão como um instrumento de interação. Em algumas aplicações, o uso da informação dos dedos é de fundamental importância durante o processo de interação, pois assim será possível executar comandos mais flexíveis e precisos não possíveis com apenas o uso de gestos da mão.

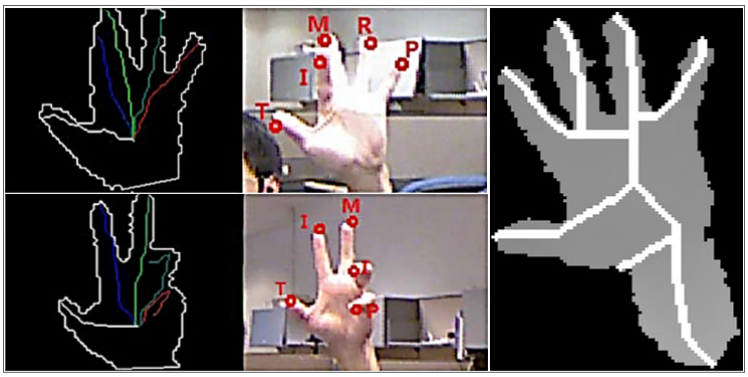


Figura 2.5: Abordagens para detecção dos dedos utilizando a distância geodésica entre a ponta do dedo e o centro da mão. a) imagem indicando o contorno da mão com os caminhos da distância calculados juntamente com a imagem RGB dos dedos detectados e rotulados (LIANG *et al.*, 2012); b) imagem contendo o caminho da distância entre as pontas dos dedos e o centro da mão (KREJOV; BOWDEN, 2013). Figuras extraídas de Liang *et al.* (2012) e Krejov e Bowden (2013), respectivamente.

Vários tipos de abordagens têm surgido com o intuito de melhor atender as necessidades exigidas nas mais diversas aplicações levando em consideração as suas limitações. As principais abordagens aqui apresentadas são classificadas da seguinte maneira: distância, contorno e aparência.

2.2.1 Análise da distância

Uma maneira interessante para detectar os dedos é analisando a estrutura da malha da mão, utilizando como princípio a distância entre os pontos de interesse presentes na estrutura da mão.

Um detector robusto é apresentado por Liang *et al.* (2012), onde se propõe uma abordagem fundamentada no algoritmo de extração denominado Acumulativo da Geodésica Extrema (AGEX), em que os dedos são considerados os pontos que maximizam a distância geodésica a partir do centro da palma da mão para as extremidades (FIGURA 2.5-a).

O trabalho desenvolvido por Krejov e Bowden (2013) analisa a superfície da mão utilizando a distância geodésica para determinar a sua extremidade. Os resultados obtidos são encorajadores, porém, demanda um elevado custo computacional devido a utilização de algoritmos como *Dijkstra* (DIJKSTRA, 1959).

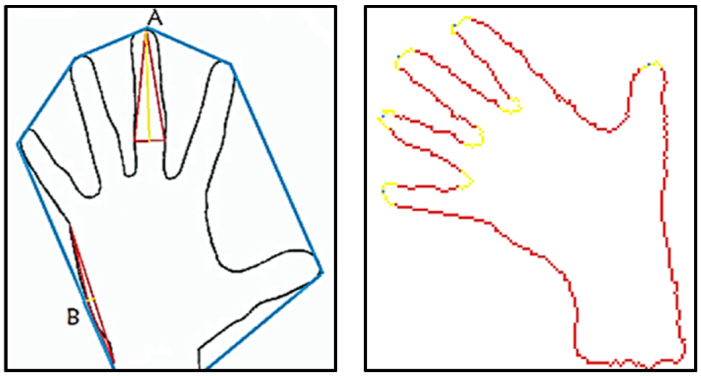


Figura 2.6: Detecção da mão analisando o contorno. a) abordagem apresentada por Wen *et al.* (2012), onde é feita uma análise do defeito da convexidade com triângulos isósceles indicando a ponta do dedo (I) e o caso contrário (II); b) análise das curvas do contorno para detectar a ponta dos dedos, região em vermelho indica o contorno da mão, regiões em amarelo indicam os pontos da curva e os pontos azuis indicam a localização da ponta dos dedos (RYAN, 2012). Figuras extraídas de Wen *et al.* (2012) e Ryan (2012), respectivamente.

2.2.2 Análise do contorno

O uso do contorno da mão como premissa para detecção dos dedos tem sido uma estratégia simples e promissora, uma vez que a disposição dos dedos pode facilitar a detecção.

A proposta de Wen *et al.* (2012) utiliza uma técnica com base na análise dos vértices extraídos pelo algoritmo de convexidades sobre o contorno da mão. Considerando a característica morfológica do dedo, cada vértice é considerado como a ponta do dedo se e somente se: estes juntamente com os outros dois pontos do contorno da mão formam um triângulo isóscele, e satisfaça uma condição sobre o ângulo formado pelo vértice no triângulo, bem como a sua altura em relação à base (FIGURA 2.6-a).

O trabalho de Ryan (2012) consegue predizer a localização das pontas dos dedos bem como o ponto de intersecção entre os dedos utilizando o método de Curvatura-K para detectar curvas ao longo do contorno da mão (FIGURA 2.6-b).

Embora apresente suas vantagens, esta abordagem é limitada pela necessidade dos dedos estarem sempre estirados de maneira que estes componham parte da silhueta da mão, consequentemente reduzindo a capacidade do método em atender as mais variadas posturas da mão.

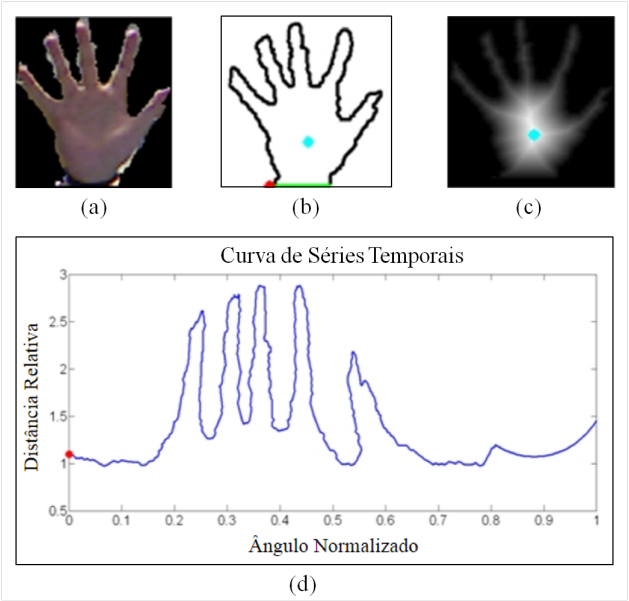


Figura 2.7: Abordagem para a detecção da mão utilizando o método de Curva de Séries Temporais proposta por Ren *et al.* (2011). a) a imagem RGB segmentada; b) o contorno da mão com o seu centro em azul e definição da área do pulso em verde; c) imagem da transformada da distância com o seu centro. d) curva das Séries Temporais referente ao contorno da mão (imagem *b*). Figura extraída de Ren *et al.* (2011).

2.2.3 Análise da aparência

O trabalho de Ren *et al.* (2011) utiliza os dados da aparência para detectar a mão FIGURA 2.7-a e então representa o contorno desta utilizando o método de Curva de Séries Temporais para posteriormente detectar os dedos utilizando a abordagem baseada no esquema de decomposição de forma quase convexa (*Near-convex*) (FIGURA 2.7-d).

Utilizando o conceito de correspondência entre modelos, Oka *et al.* (2002) utilizam o método de correlação normalizada juntamente com um modelo de um círculo para detectar a ponta dos dedos, considerando a forma global do dedo como um cilindro de formato esférico na extremidade. Devido ao alto grau de liberdade da mão, abordagens que utilizam a aparência podem afetar consideravelmente os resultados devido ao critério de generalização, ou seja, este tipo de abordagem é sensível às alterações na postura que a mão pode assumir.

Tendo como objetivo reconhecer caracteres utilizando o conceito de Escrita-no-Ar e levando em consideração que o dedo estará sempre à frente do corpo e apontando para a câmera, Feng *et al.* (2012) fazem uma análise sobre a região da mão e parte do braço, e posteriormente são segmentados pelo método de K-médias para indicar que a ponta do dedo é o ponto mais afastado da região do braço.

A detecção dos dedos apresentada por Raheja *et al.* (2011) é resultado da subtração da região da palma com a região inteira da mão, ambas extraídas previamente utilizando o mapa de profundidade. Assim, a ponta do dedo é considerada como a região que apresenta o mínimo valor de profundidade em cada dedo.

2.3 Rastreamento da mão

O objetivo do processo de rastreamento é predizer a trajetória de um objeto sobre um determinado período, inferindo a sua posição a cada quadro. O rastreamento da mão, em particular, é considerado complexo, pois a aparência da mão poderá mudar caso esta mova rapidamente.

2.3.1 Rastreamento baseado em aparência

O uso de informações tridimensionais, forma e borda foram por muito tempo a base para diversas abordagens em rastreamento da mão e ainda são. Contudo, apresentam limitações devido à sensibilidade às condições de iluminação do ambiente, bem como a dificuldade em explorar as informações presentes no espaço tridimensional. Atualmente, devido principalmente à crescente facilidade em capturar e interpretar as informações 3D da cena (Microsoft Kinect, Asus Xtion Pro, etc), os dados de profundidade têm sido amplamente explorados no contexto de rastreamento em geral, principalmente da mão.

Considerado uma evolução do método *Mean Shift* (MS), o método *Continuously Adaptive Mean Shift* (CAMShift) adicionalmente adapta dinamicamente o tamanho da região a ser rastreada, sendo assim robusto em situações que há variações na escala. Ambos são apropriados para o rastreamento de objetos deformáveis (SUAREZ; MURPHY, 2012), mas não apresentam bons resultados quando aplicados sobre os dados de profundidade (HONGYONG; YOULING, 2012), visto que estes são baseados na histogramização de regiões da imagem e podem causar falsos positivos para objetos situados na mesma faixa de profundidade do objeto a ser rastreado. Como possível solução, Yang *et al.* (2012) propõem uma abordagem otimizada para o rastreamento da mão utilizando o CAMShift, onde é realizado um cálculo de Probabilidade Ponderada Gaussiana da imagem de profundidade em relação à ponta da mão. Os resultados alcançados foram utilizados para controlar um sistema de tocador de mídia, sendo executado a uma velocidade de 14 quadros por segundos.

Utilizando o Filtro de Kalman (FK)[2], Tang *et al.* (2012) propõem rastrear o centroide da mão no espaço tridimensional. Em casos de sobreposição das mãos, o algoritmo de crescimento de regiões é utilizado para extrair as duas regiões sobrepostas. Uma implementação robusta e detalhada do FK para rastreamento da mão em imagens de profundidade foi desenvolvida por Park *et al.* (2012). Krejov e Bowden (2013), utilizam o FK para rastrear e atribuir simultaneamente a ponta de cada dedo em um espaço tridimensional entre as imagens.

Com o objetivo de simular uma mão virtual, Vicente e Faisal (2013) fazem uso de marcadores coloridos sobre a ponta dos dedos para facilitar o processo de rastreamento, bem como determinar a posição e orientação da mão virtual. O vetor de orientação da mão são os autovetores calculados por meio da Análise de Componentes Principais (ACP)[3] de todos os pontos referentes a região da mão no espaço tridimensional. Com a posição inicial da ponta de cada dedo, Liang *et al.* (2012) implementam um filtro de partícula para rastrear tais posições através das imagens sucessivas.

O trabalho apresentado por Hongyong e Youling (2012) propõe um rastreamento por bolhas (*blobs*) utilizando o algoritmo dos K-vizinhos mais próximos (KVP)[4]. As mãos ou dedos presentes na imagem de profundidade segmentada são tratados como bolhas e é realizada uma busca por regiões semelhantes entre duas imagens consecutivas analisando os KVPs. A partir das informações obtidas foi possível identificar gestos, os quais foram utilizados para controlar um teclado virtual e realizar manipulações simples sobre um objeto virtual bidimensional.

A proposta apresentada por Li *et al.* (2012) é baseada em aprendizado estatístico, onde é apresentado um conjunto modificado de características baseadas nas características de *Haar*, capazes de identificar os padrões de variações da mão em imagens de profundidade. Apesar de apresentar bons resultados, possui baixa robustez quando a mão está sobre o corpo ou sobre algo cuja forma é semelhante a mão.

O trabalho desenvolvido por Tang *et al.* (2015) propõe uma abordagem robusta para detectar e rastrear a mão utilizando imagens rgb e de profundidade capturadas pelo sensor Kinect. Adicionalmente, utiliza o método de *Deep Belief Networks* (DBN) para aprender características da postura da mão insenssíveis ao movimento, escala e rotação, alcançando uma taxa de exatidão igual a 98,12%.

[2]KF - Kalman Filter.
[3]PCA - Principal Component Analysis.
[4]KNN - K-Nearest Neighbors.

2.3.2 Rastreamento baseado em modelo

Adicionando informação de profundidade à técnica de eixo medial, Ouedraogo e Aoki (2014) conseguiram estimar e rastrear as posições tridimensionais das articulações sem a necessidade de aplicar classificadores para estimar a postura da mão.

O trabalho apresentado por Samadani *et al.* (2012) propõe um modelo cinemático inverso multi-restritivo sendo possível estimar rapidamente e precisamente a configuração das juntas para uma determinada pose da mão. É abordado um paradigma de alta e baixa prioridade de restrição das juntas, sendo adequado para auxiliar métodos de captura de movimento baseados em marcadores, pois podem apresentar ausência de dados (oclusão, deslocamento) de algum marcador durante a sua execução.

A abordagem desenvolvida por Qian *et al.* (2014) alcançou um resultado muito promissor e aproveitou técnicas baseadas no rastreamento do corpo humano. Os autores sugerem um novo modelo tridimensional para estimar a cinemática da mão inspirado no trabalho de Oikonomidis *et al.* (2010). Porém, este modelo cinemático é definido como um conjunto de 48 esferas possuindo 26 graus de liberdade, as quais são projetadas dentro de uma nuvem esparsa de pontos tendo uma função de custo para medir a discrepância. Todo o sistema alcançou uma velocidade de 25 quadros por segundos e demonstrou ser robustos durante os testes.

Segundo Tompson *et al.* (2014), estudos realizados com redes de convolução foram conduzidos para recuperar pose de objetos (rígidos e não rígidos), rostos e partes do corpo humano apresentados em LeCun *et al.* (2004) e Osadchy *et al.* (2005). Adicionalmente, Jiu *et al.* (2014) reconheceram poses do corpo humano. Entretanto, Tompson *et al.* (2014) demonstraram ser os pioneiros a reproduzir esta técnica para recuperar continuamente a pose tridimensional da mão humana usando dados de profundidade. Como resultado, apresenta uma taxa de erro de 4,1% para o conjunto de dados de validação, apresentando falhas quando a mão é ocluída com alguma parte do corpo.

O trabalho conduzido por Schroder *et al.* (2014) também encaixa um modelo tridimensional em uma nuvem de pontos capturados pelo sensor Kinect. Contudo, reduz o espaço de representação da postura da mão por analisar a sinergia desta e apenas relevar os relacionamentos entre suas partes (por exemplo, os dedos) que definem a postura. Mesmo com os dados reduzidos, uma análise demonstra que utilizar a nuvem de pontos completa pode acarretar em falhas durante o rastreamento. A elevada presença de pontos em determinado segmento da nuvem pode gerar uma pseudo interpretação (por exemplo, o colapso de dedos do modelo tridimensional quando os dedos da mão estiverem muito próximos um do outro). Esta abordagem alcançou uma velocidade de execução de 30 quadros por segundo.

2.4 Relação com o trabalho proposto

A detecção da mão na imagem é o passo fundamental para as diversas tarefas envolvendo o reconhecimento dos gestos. Várias estratégias tem sido abordadas para melhor executar este procedimento como visto nas Seções 2.1 a 2.4. Utilizando o conceito das INUs, o sensor Kinect foi posicionado verticalmente com a intenção de focar nas mãos do usuário que esteja sentado em frente ao computador com os braços ligeiramente estendidos para frente.

2.4.1 Detecção da mão

Semelhante ao trabalho desenvolvido por Liang *et al.* (2012), o nosso trabalho utiliza uma segmentação por limiar para a extração do primeiro plano juntamente com uma análise feita sobre a região da mão. A proposta é detectar a mão de maneira rápida e eficaz sem demandar um alto custo computacional. Por isso, foi proposto um método de correspondência de modelo para verificar se a mão está situada em uma região pré-definida da imagem. Este processamento foi realizado sobre uma imagem binária, utilizando como modelo a forma da mão aberta perpendicular ao campo de visão da câmera.

As imagens capturadas pelo nosso trabalho apresenta um plano de fundo menos complexo devido ao posicionamento da câmera de forma vertical à mesa em que está situado o computador, pois o campo de visão sempre apontará para um plano, facilitando o processo de detecção da mão. Embora apenas a informação de profundidade tenha sido utilizada para realizar a detecção da mão neste trabalho, a maneira de como a câmera foi configurada descarta a necessidade de realizar complexas segmentações por cor da pele como nos trabalhos apresentados por Bergh e Gool (2011) e Xu *et al.* (2011).

A abordagem proposta por Lee *et al.* (2016) utiliza as imagens em profundidade da mão dividias em blocos não sobreposto, onde a porcentagem dos pixels e média de profundidade em cada bloco são utilizadas como características para treinar um classificador baseado nas Máquinas de Vetores de Suporte (MVS)[5]. Semelhante ao nosso trabalho, propomos um método para extrair características sobre os dados de profundidade para detectar a mão baseada no processo de correspondência entre descritores.

[5]SVM - Support Vector Machine.

2.4.2 Rastreamento da mão

O objetivo do rastreamento é localizar a mão na imagem em uma sequência de quadros sucessivos para posteriormente realizar algum processamento sobre a postura da mão. A técnica de rastreamento utilizada nesta pesquisa é baseada na detecção da mão em imagens sucessivas utilizando como base o algoritmo *Oriented FAST e Rotated BRIEF* (ORB) (RUBLEE *et al.*, 2011) em paralelo com a Transformada da Distância Euclidiana (TDE)[6] (BORGEFORS, 1986). O ORB foi criado a partir da fusão dos métodos *Features from Accelerated Segment Test* (FAST) (ROSTEN; DRUMMOND, 2006) implementado de forma invariante à rotação, e o *Binary Robust Independent Elementary Features* (BRIEF) (CALONDER *et al.*, 2010) com algumas modificações, sendo considerado uma alternativa para os métodos de *Scale-Invariant Feature Transform* (SIFT) (LOWE, 2004) e *Speeded Up Robust Features* (SURF) (BAY *et al.*, 2008), onde é necessário o uso de modelos para encontrar o objeto (mão) correspondente em outra imagem.

O método de CAMShift apresenta algumas particularidades que influenciam diretamente no desempenho do rastreador, pois este tem bom desempenho em objetos que tenham uma aparência simples e constantes, mas não é robusto quando o plano de fundo apresenta cor similar ao objeto alvo (RAUTARAY; AGRAWAL, 2012). Embora apresente baixo custo computacional, é necessário extrair características que identifiquem a mão para iniciar o rastreamento, comumente utilizando dados de cor (YIN *et al.*, 2009).

[6] EDT - Euclidean Distance Transform.

Capítulo 3

Detecção e Rastreamento das Mãos

Conteúdo

3.1 Visão Geral

O método implementado detecta a mão em imagens de profundidade e cria um modelo cinemático para esta. Este trabalho engloba, em sua maioria, técnicas clássicas de processamento de imagens para compor a solução final do problema. Contudo, alguns algoritmos foram propostos como alternativas para contornar dificuldades encontradas e sobretudo viabilizar a concretização do mesmo.

O método foi dividido em quatro principais estágios: pré-processamento, extração e seleção das características, rastreamento da mão, modelo cinemático. Estes processos são subsequentes, ou seja, os dados processados a cada estágio servirão de parâmetro para o estágio seguinte, exceto o procedimento da coleta de dados, o qual é executado apenas uma vez extraindo características para o rastreamento. A visão geral de toda aplicação pode ser acompanhada abaixo.

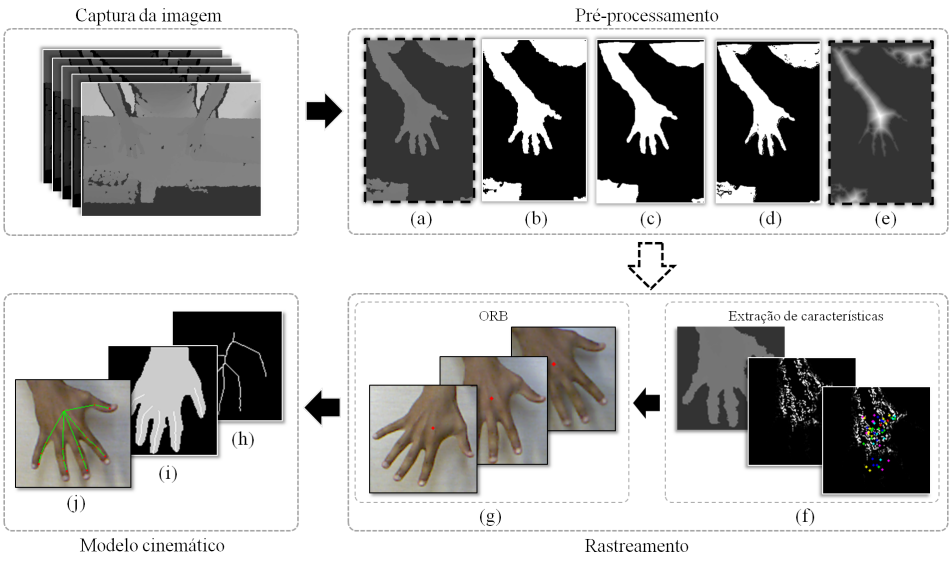


Figura 3.1: Visão geral do método proposto dividido em três grandes etapas: Pré-processamento ((a) imagem de profundidade; (b) imagem de profundidade segmentada e binarizada; (c) imagem binária suavizada; (d) imagem binária com bordas; (e) imagem da TDE com bordas); Rastreamento: ((f) extração e seleção das características; (g) correspondência entre descritores utilizando o algoritmo ORB); Modelo Cinemático ((h) imagem da mão após afinamento; (i) eixos medianos de cada dedo; (j) modelo cinemático final). A imagem *(a)* e *(e)* são utilizadas no estágio de rastreamento (seta tracejada), sendo utilizadas na extração de características e aplicação do rastreamento, respectivamente.

Captura da imagem

O sensor Kinect foi o equipamento de captura utilizado e teve sua posição definida de maneira que abrangesse apenas a região de atuação das mãos, visando detectar apenas a configuração (orientação e posicionamento dos dedos) desta. Assim, tal sensor foi posto paralelamente à linha do horizonte, sendo o campo visual formado pelas mãos de uma pessoa ao estendê-las para frente. Independente da aplicação utilizada para este método, seja para sistemas de cinemetria (ABREU, 2013) ou para interação com sistemas computacionais, o modo de captura deve ser restrito a este campo visual, uma vez que o intuito é extrair informações particulares da mão e seu movimento, além de ser mais confortável para a realização dos gestos (FIGURA 3.2).

Pré-processamento

O sensor Kinect foi lançado pela Microsoft em 2010 e foi amplamente aceito pela comunidade por ser um equipamento de baixo custo capaz de gerar imagens de profundidade com uma qualidade aceitável (ANDERSEN *et al.*, 2012). No entanto, este sensor utiliza o princípio da luz estruturada para o cálculo do valor da profundidade (KEAN *et al.*, 2011) e, portanto, a natureza de captura dos dados pode produzir ruídos (ausência de dados,

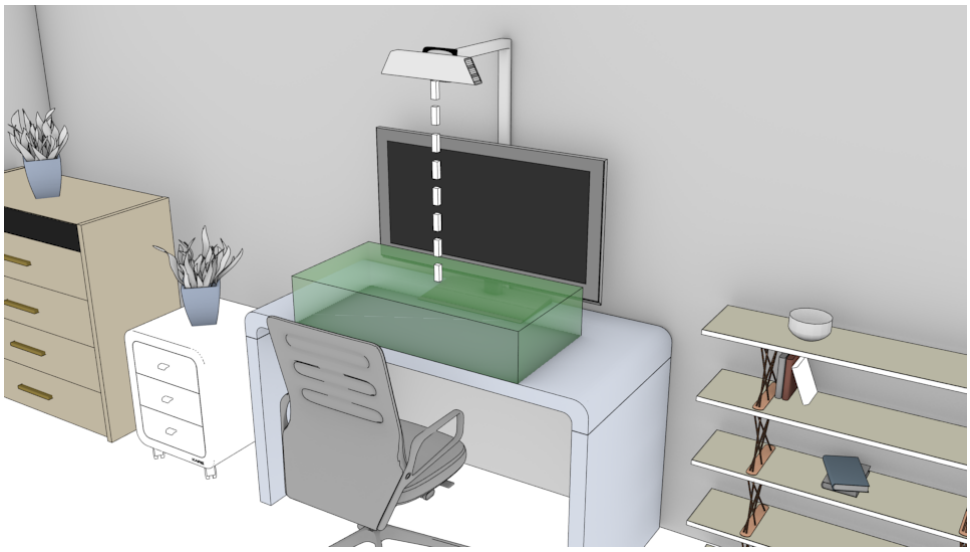


Figura 3.2: Configuração da posição do sensor Kinect para aplicação do método em sistemas computacionais. A linha tracejada entre o Kinect e a mesa tem uma altura de 0,9 metros (aproximadamente) e em conjunto com o paralelepípedo verde sobre a mesa (0,95 x 0,45 x 0,11 metros) indicam a área de atuação das mãos.

dados inconsistentes e/ou oclusão) em determinadas regiões: em superfícies reflexivas (FIGURA 3.3-a), nas bordas de objetos (FIGURA 3.3-b) e em objetos muito próximo ao sensor (FIGURA 3.3-c). Em vista disto, a utilização de dados advindos do Kinect deve ser tratada em um passo anterior ao processamento. A etapa de pré-processamento adotada neste projeto engloba operações de transformações em imagens (corte e inversão), conversão de tipo (8, 16 e 32 bits), segmentação (remoção do plano de fundo), suavização e detecção de bordas (Acúmulo da Diferença da Vizinhança mais próxima - ADV4) e a TDE. Todo o processamento contido nesta etapa (FIGURA 3.1) é descrito detalhadamente na subseção 3.2.1.

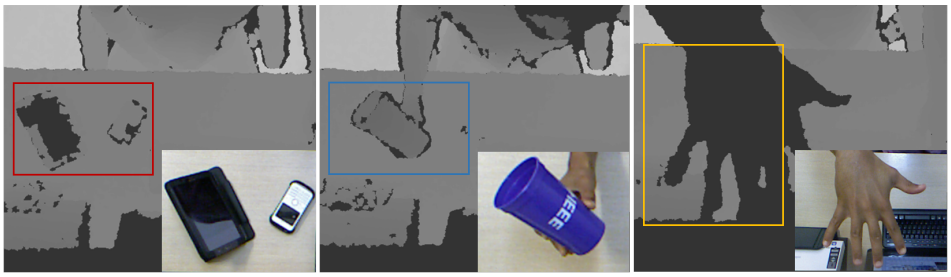


Figura 3.3: Exemplos de ruídos presentes na imagens de profundidade gerada pelo sensor Kinect. a) ausência de dados em regiões reflexivas (retângulo vermelho); b) inconsistência de dados em regiões próximas às bordas de objetos (retângulo azul); c) objetos próximos à câmera bloqueiam a propagação dos raios infravermelhos emitido pelo sensor, gerando assim a oclusão (retângulo amarelo).

Extração e seleção de características

Após o pré-processamento é aplicado o algoritmo para rastrear a mão comparando os descritores locais da mão na imagem da cena e alvo, assim, a extração e seleção de características foi criada para extrair e armazenar as características (pontos chaves) da mão, na imagem alvo, que poderão melhor descrevê-la. Todo o processamento contido nesta seção não influência no custo computacional final do método proposto, pois é executado à parte com o intuito apenas de coletar características. A extração dos pontos chaves da mão na cena e comparação entre descritores, compõe o estágio de detecção, sendo o rastreamento o processo contínuo desta detecção.

A fim de extrair pontos que melhor descrevam a imagem da mão, foi introduzido um algoritmo para extração de característica baseado no volume do retângulo formado pelo vetor normal no espaço de coordenadas esféricas. Planos tangentes pertencentes à um objeto, contém, naturalmente, informações sobre a sua superfície. Tais planos são, comumente, tratados como vetores normais. Dados dois vetores tangentes à um ponto pertencente a um plano tangente, o vetor normal pode ser definido pelo produto vetorial deste vetores (PISSANETZKY, 2004). A figura 3.8 exibe um espaço de coordenadas esféricos com os ângulos de inclinação de um vetor normal (azimute e zênite) e sua distância r. Diferentes valores para r implicam em novos valores para os eixos x, y *e* z. A característica é definida calculando o volume do retângulo formado pelo vetor normal considerando valores arbitrários para r.

O algoritmo ORB foi utilizado para localizar e descrever as características locais, sendo proveniente da fusão do detector de pontos chaves FAST (ROSTEN; DRUMMOND, 2006) e o descritor BRIEF (CALONDER *et al.*, 2010). A imagem resultante do pré-processamento foi utilizada para extração dos pontos chaves. Posteriormente, foi realizada uma seleção desses pontos com base nos seus respectivos índices de confiança. Ao todo foram utilizadas 5000 imagens da mão (direita e esquerda) divididas em cinco categorias (poses) diferentes. Para cada pose foram utilizadas 500 imagens, cada qual gerando um conjunto de pontos. O conjunto final é composto pelos pontos de todas as poses.

Rastreamento da mão

A técnica de rastreamento (detecção contínua da mão) utilizada é baseada na comparação de descritores locais de duas imagens diferentes. Os dados coletados previamente compõem os pontos chaves da mão na imagem alvo e durante o rastreamento ocorre a extração destes pontos na cena capturada. O método de Consenso entre Amostras

Aleatórias (CAA)[1] (FISCHLER; BOLLES, 1981) foi utilizado para selecionar apenas os pontos com uma taxa de correspondência alta. Por fim, foi implementado um método de seleção por bolhas para localizar a posição da mão considerando o número de pontos correspondidos por pixel quadrado.

Modelo cinemático

O modelo cinemático da mão é constituído por pontos interligados por linhas que representam a posição e articulações das falanges de cada dedo. Inicialmente, o algoritmo de afinamento (ZHANG; SUEN, 1984) é aplicado sobre a imagem binária da mão (120x120 pixels) para detectar os eixos medianos de cada dedo (linhas dos dedos) e, consequentemente, definir o seu respectivo tamanho. Então, utilizando as proporções dos segmentos da mão (BURYANOV; KOTIUK, 2010) é possível localizar o centro de cada falange. Finalmente, o modelo cinemático é formado e impresso sobre a imagem RGB capturada. Todo este procedimento é descrito na subseção 3.2.4.

Ambiente de desenvolvimento

O framework foi desenvolvido em linguagem *C++* e teve como auxílio as bibliotecas *Open Natural Interaction* (OpenNI) (PRIMESENSE, 2016) e *Open Computer Vision* (OpenCV) (BRADSKI, 2000) para realizar a captura e processamento das imagens, respectivamente. Os resultados analisados foram realizados em uma máquina com processador Core i5 de 2,5 giga-hertz e com memória *RAM* de 3 gigabytes.

3.2 Método

Esta seção apresenta, detalhadamente, os métodos e soluções descritos na seção anterior. Conceitos clássicos da literatura tais como a segmentação, suavização e detecção de bordas, correspondência entre descritores locais são explorados e compõem grande parte do conjunto de abordagens. Cada subseção descreve o procedimento adotado em cada etapa, ressaltando a sua importância dentro do sistema proposto.

3.2.1 Pré-processamento

A tarefa de gerenciamento de gestos da mão utilizando Visão Computacional exige um equilíbrio entre complexidade e eficiência, uma vez que, os tempos de análise e

[1] RANSAC - Random Sample Cosensus.

resposta devem ser suficientemente rápidos para a realização de uma interação homem-máquina. Técnicas para melhorar a eficiência destes procedimentos são fundamentais para alcançar resultados satisfatórios para a aplicação do trabalho. Esta seção destina-se a apresentar as técnicas e estratégias utilizadas para tratar e melhorar a qualidade da imagem sem comprometer o desempenho. O resultado final após este procedimento será um conjunto de imagens pré-processadas que são utilizadas nos estágios de coleta de dados, rastreamento da mão e detecção da cinemática da mão.

Definição da região de interesse

O Kinect gera um mapa de profundidade de resolução VGA (640×480 pixels) a uma frequência de 30 Hz. Devido à maneira como o Kinect foi posicionado para capturar as imagens das mãos, as regiões extremas da imagem não são exploradas, sendo a parte central o foco para análise do rastreamento. Para amenizar o custo computacional durante a fase de processamento, a imagem de profundidade foi cortada e sua dimensão reduzida para um tamanho de 580×430 pixels. Adicionalmente, os braços e as mãos posicionados à frente do sensor Kinect ficam evidentes no centro da imagem facilitando a aplicação do método. Foi necessário inverter a imagem horizontalmente para tornar o ponto de captura da imagem similar a perspectiva do usuário (FIGURA 3.4).

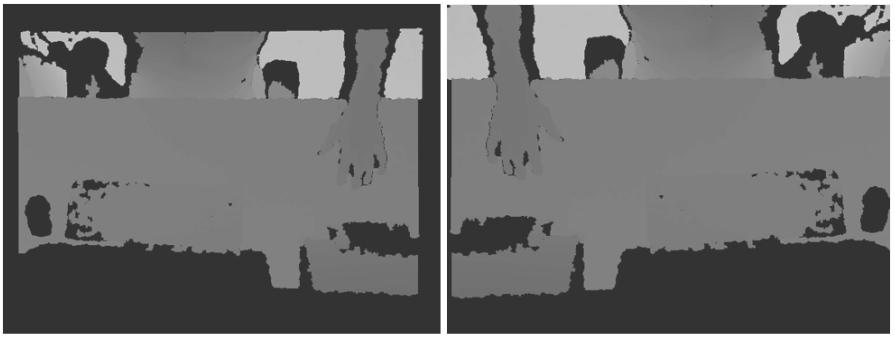
Conversão de tipo

Cada pixel do mapa de profundidade é representado por um valor de 16 bits (MILES, 2012), sendo considerados válidos apenas os valores entre 0 e 2047 (11 bits) (HONGYONG; YOULING, 2012). Ao longo da pesquisa, foi necessário convertê-los para outras profundidades de cores: 8 bits para gerar imagens binárias e 32-bits para operações com pontos flutuantes, como a criação da TDE, bem como o cálculo do vetor normal durante o processo de extração de características.

Segmentação da mão

O processo de remoção do plano de fundo em imagens é uma tarefa inerente a abordagens de detecção e rastreamento de objetos em um cenário, sendo fundamental para identificar as regiões estáticas e regiões em movimento presentes nestas (SANCHES *et al.*, 2013).

Em mapas de profundidade, o uso de limiares para segmentar o plano de fundo é uma tarefa muito comum devido a eficácia alcançada com baixa complexidade computacional.



(a) (b)

Figura 3.4: Definição da região de interesse. a) imagem de profundidade gerado pelo Kinect convertida para 8-bits (640 × 480 pixels); b) imagem *(a)* cortada (580 × 430 pixels) e invertida horizontalmente.

Tal procedimento é utilizado para definir uma região fixa no cenário para análise dos gestos, descartando áreas irrelevantes (regiões estáticas) ao processamento. Assim, foram selecionados dois limiares levando em consideração o limite mínimo e máximo de captura do Kinect. Seja D, a imagem de profundidade adquirida pelo Kinect; *(x,y)*, as coordenadas de um pixel (P_i) e considerando, L_{ant} e L_{pos}, os limiares utilizados para definir o início e fim da faixa de segmentação, respectivamente, a imagem segmentada (D_{seg}) pode ser obtida pela Equação 3.1:

$$D_{seg}(x,y) = \begin{cases} P_i & \text{para } L_{ant} \leq P_i \leq L_{pos} \\ 0 & \text{para outro caso} \end{cases} . \tag{3.1}$$

A menor e a maior distância possível que o kinect consegue determinar com um grau de precisão aceitável varia de 0,5 a 5 metros, respectivamente (KHOSHELHAM; ELBERINK, 2012). Levando em consideração que os valores próximos a 0,5 metros são susceptíveis à apresentar maior incidência de ruídos e considerando a banca da mesa (plano de fundo) à 0,9 metros (FIGURA 3.5-a), os valores para os limiares L_{ant} e L_{pos} foram 0,72 e 0,83 metros, respectivamente. Ambos foram determinados por observar a imagem e verificar o valor mínimo de profundidade que não incorra em insidência de ruídos sobre à mão quando estiver na cena (L_{ant}), e paralelamente, o maior valor de profundidade possível que descarte o plano de fundo (FIGURA 3.5-b). As porcentagens destes limiares em relação à altura do sensor (80% e 92%, respectivamente) poderão ser utilizados caso ocorra alteração na altura deste. A imagem binária (D_{bin}) é o produto final após o processo de segmentação (FIGURA 3.5-c), sendo definida pela Equação 3.2:

$$D_{bin}(x,y) = \begin{cases} 255 & \text{para } D_{seg}(x,y) \neq 0 \\ 0 & \text{para outro caso} \end{cases} , \tag{3.2}$$

Figura 3.5: Imagens das etapas de segmentação; a) mapa de profundidade (8 bits); b) mapa de profundidade segmentado; c) imagem binária após segmentação do mapa de profundidade (D_{bin}).

Suavização do contorno

O contorno de objetos presentes no mapa de profundidade pode não representar a forma correta devido a elevada instabilidade dos pixels presentes nestas áreas. Assim, tal situação pode influenciar de forma negativa na aplicação de técnicas que são sensíveis à detecção de bordas (FIGURA 3.6-a).

Para suavizar o contorno dos objetos foi aplicado o filtro da média (JAIN *et al.*, 1995). Seja i e j, as coordenadas de um pixel na imagem binária (D_{bin}); l e m, as coordenadas dos seus vizinhos, e k o tamanho do kernel utilizado, a imagem suavizada (S_{bin}) foi obtida pela equação 3.3. Posteriormente foi realizada a operação morfológica de erosão (FISHER *et al.*, 1996) para amenizar o efeito de dilatação causado, cujo o tamanho do elemento estruturante E é exibido na equação 3.4. Ao final, imperfeições da região do contorno são suavizadas, conforme a figura 3.6-b.

$$S_{bin}(i,j) = \begin{cases} 255 & \text{para } \left(\frac{1}{(k=6)} \sum_{l=i-1}^{i+1} \sum_{m=j-1}^{j+1} D_{Bin}(l,m)\right) > 0 \\ 0 & \text{para outro caso.} \end{cases} , \tag{3.3}$$

$$E = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{3.4}$$

Detecção das bordas

As bordas indicam a transição entre duas regiões em uma imagem, podendo representar descontinuidades nas superfícies, profundidade entre objetos e/ou superfícies de cor ou textura (KRIM; HAMZA, 2015), podendo assim armazenar informações importantes sobre a superfície de um objeto.

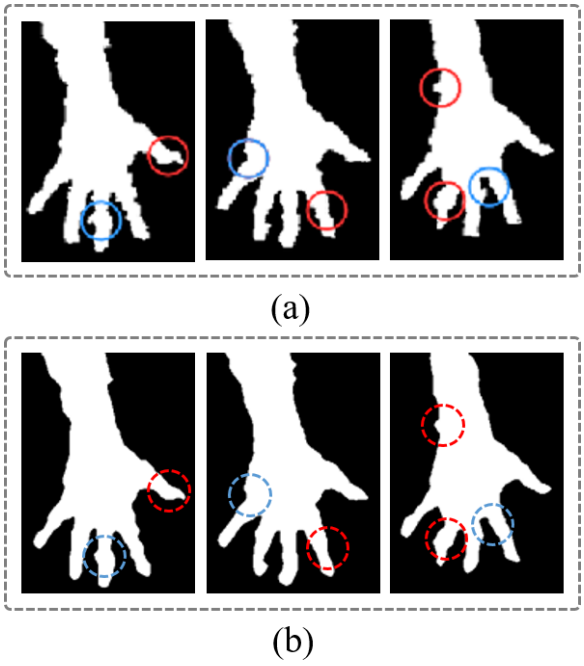


Figura 3.6: Processo de suavização do contorno. a) ruídos presentes no contorno da mão na imagem binária, onde os círculos vermelhos indicam regiões que estão em excesso e os azuis indicam regiões com ausência de dados. b) imagem após a suavização do contorno, círculos tracejados indicam as imperfeições suavizadas.

A imagem binária suavizada (S_{Bin}) é utilizada para gerar a imagem da TDE e posteriormente será usada no rastreamento. Porém, uma imagem binária contém baixa informação sobre a superfície de um objeto (apenas o contorno), sendo sensível à variação quanto ao ponto de vista. Assim, para aumentar o poder discriminativo do rastreador, foi necessário adicionar a informação de borda e ressaltar as variações de profundidade presentes no interior do contorno, bem como as nuances da superfície. Para isto, foi proposto um algoritmo para detectar bordas baseado no acúmulo das diferenças entre um pixel e seus vizinhos mais próximo denominado aqui de (ADV-4).

Seja *D(x,y)* um pixel na imagem de profundidade (valores normalizados entre 0 e 255 com 32 bits de profundidade em precisão simples) e, x e y suas coordenadas e i o índice para o cálculo das coordenadas dos pixels vizinhos mais próximos (-1 à 1), a imagem (D_{Borda}) após a aplicação do filtro é definida por:

$$D_{Borda}(x,y) = \sum_{i=-1}^{1} (D(x,y) - D(x+i,y)) + (D(x,y) - D(x,y+i)) \tag{3.5}$$

A imagem D_{Borda} (FIGURA 3.7-c) é então binarizada de acordo com a equação 3.2 e processada com o filtro de negativo para posteriormente aplicar seus valores sobre a imagem S_{Bin}. Então, a imagem S_{Bin} alterada (FIGURA 3.7-d) é utilizada para criar a

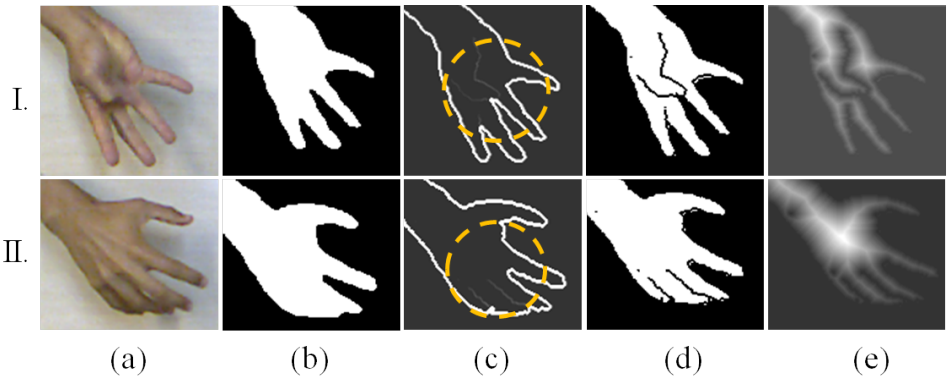


Figura 3.7: Ilustração do processo de detecção de bordas em duas poses. a) imagem RGB capturada; b) S_{Bin} - imagem binária suavizada e gerada a partir do mapa de profundidade segmentado (D_{Seg}); c) imagem após o detector de bordas gerada a partir do mapa de profundidade normalizado (D_{Borda}). Círculos amarelo indicam regiões onde há presença de bordas internas; d) imagem binária (*b*) com as bordas; e) imagem da TDE com bordas (D_{BTDE}) gerada a partir da imagem *(d)*.

imagem da TDE (FIGURA 3.7-e) com bordas, sendo definida como D_{BTDE}. Para o cálculo da TDE foi utilizada a abordagem sugerida por Borgefors (1986), considerando *a = 0,955* (vizinhos mais próximos), *b = 1,36930* (vizinhos da diagonal) e tamanho da vizinhança igual a 3×3 pixels. Todo o processo de detecção é visualizado na FIGURA 3.7.

3.2.2 Extração das características da mão

O rastreamento proposto neste trabalho é baseado na comparação entre descritores locais da imagem capturada e da imagem modelo do objeto a ser rastreado utilizando os dados de profundidade. No entanto, o uso individual da informação de profundidade produz descritores locais com baixo poder discriminativo. Para contornar este problema foi aplicado o procedimento de excesso de descrição, onde é possível exceder a descrição de uma característica por combinar diferentes tipos de valores (cor RGB, gradiente da intensidade de cor local, profundidade) e é explicado por Krig (2014).

A técnica de representação de característica proposta neste trabalho, denominada Volume da Normal (VNOR), visa melhorar a descrição das características no mapa de profundidade utilizando dois tipos de dados: o volume do vetor normal de cada pixel convertido para o sistema de coordenadas esféricas (FIGURA 3.8) considerando em seu cálculo a informação da distância Euclidiana em relação a borda mais próxima (TDE). As seções abaixo apresentam o método utilizado para extrair e selecionar as características após a aplicação do VNOR. Todo o procedimento é realizado sobre a imagem objeto (imagem de profundidade da mão com dimensão de 120×120 pixels). Ao final, todos os pontos coletados são armazenados e reutilizados para extração e correspondência entre os descritores na subseção 3.2.3.

Volume da normal

Uma superfície pode ser representada por um conjunto de planos tangentes à cada ponto contido nesta, podendo ser individualmente representados por um vetor normal (FIGURA 3.8). Cada vetor normal de um plano tangente é definido como o produto vetorial de dois vetores tangentes entre si e pertencentes a este mesmo plano.

Um vetor normal de um plano tangente em um ponto *(x, y, d(x,y))* é representado no sistema de coordenadas esféricas pela equação 3.6:

$$\vec{N} = \vec{S_x} \times \vec{S_y}. \tag{3.6}$$

Estes dois vetores podem ser encontrados utilizando a aproximação da diferença finita, conforme em Tang *et al.* (2013), seguindo as equações 3.7 e 3.8:

$$\vec{S_x} \approx \frac{1}{2}\Big(d(x+1,y) - d(x-1,y)\Big), \tag{3.7}$$

$$\vec{S_y} \approx \frac{1}{2}\Big(d(x,y+1) - d(x,y-1)\Big). \tag{3.8}$$

Para melhor representar a informação de orientação do vetor normal é necessário convertê-lo para o sistema de coordenadas esféricas *(θ, φ, 1)*, sendo representada pelo ângulo zênite (θ) e azimute (φ). O valor 1 indica o tamanho da normal a partir da origem. O cálculo da orientação destes ângulos são realizados utilizando as equações 3.9 e 3.10:

$$\theta = tan^{-1}\left(S_x^2 + S_y^2\right)^{\frac{1}{2}}, \tag{3.9}$$

$$\varphi = tan^{-1}\left(\frac{S_y}{S_x}\right). \tag{3.10}$$

O cálculo da normal descrito acima foi originalmente sugerido por Tang *et al.* (2013), sendo utilizado para extrair um vetor de características baseado na histogramização destes dois ângulos - uma variação da técnica do Histogramas de Gradientes Orientados (HGO)[2] (DALAL; TRIGGS, 2005) aplicada à imagens em profundidade. Em nosso trabalho, atribuímos valores arbitrários para o comprimento da normal e assim alteramos a dimensão do retângulo formado por esta no espaço de coordenadas esféricas.

[2]HOG - Histogram of Oriented Gradients

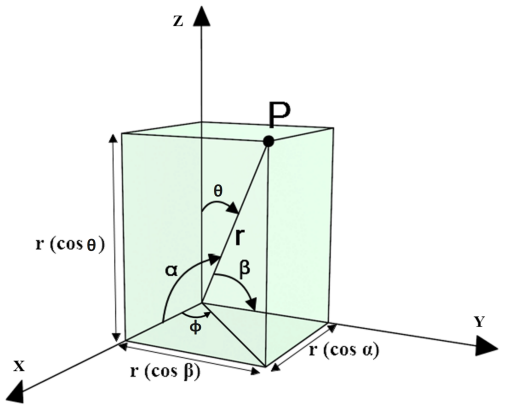


Figura 3.8: Sistema de coordenadas esféricas com um ponto **P** representando um vetor normal, sendo **r** a sua distância escalar a partir da origem. Os ângulos α, β e θ indicam a inclinação do vetor normal sobre os três eixos x, y e z, respectivamente. As equações em volta do retângulo representam o cálculo dos valores de x, y e z em relação à dimensão deste. Os ângulos θ, φ são utilizados para representar a orientação do vetor normal **P**.

Considerando P o vetor normal sobre um sistema de coordenadas esférico e utilizando a imagem D_{BTDE}, extraída na Seção 3.2.1, para atribuir valores à sua distância r, podemos redimensionar o retângulo formado pelo vetor normal (FIGURA 3.8-verde) e consequentemente determinar novos valores para os três eixos (x, y e z) de acordo com as equações 3.11, 3.12 e 3.13, conforme demonstrado em Fleisch (2008).

$$D_{BTDE}(x,y) = r \Rightarrow \begin{cases} x' = r \times \cos\alpha = r \times \sin\theta \times \cos\varphi, & (3.11) \\ y' = r \times \cos\beta = r \times \sin\theta \times \sin\varphi, & (3.12) \\ z' = r \times \cos\theta. & (3.13) \end{cases}$$

Assumindo diferentes valores para r é possível projetar diferentes retângulos sobre os três eixos e consequentemente diferentes volumes, conforme a figura 3.9. Seja x', y' *e* z', os valores correspondente a dimensão do retângulo após a atribuição dos valores a r, a imagem resultante do processo de extração de característica, D_{Caract}, é definida pela equação 3.14:

$$D_{Caract}(i,j) = (x' \times y' \times z'). \quad (3.14)$$

O processo de extração de características descrito acima encapsula duas informações sobre as superfícies dos objetos presentes na imagem: o espaço contido e orientação de cada normal (volume e orientação do retângulo da normal), bem como informações sobre o contorno da TDE.

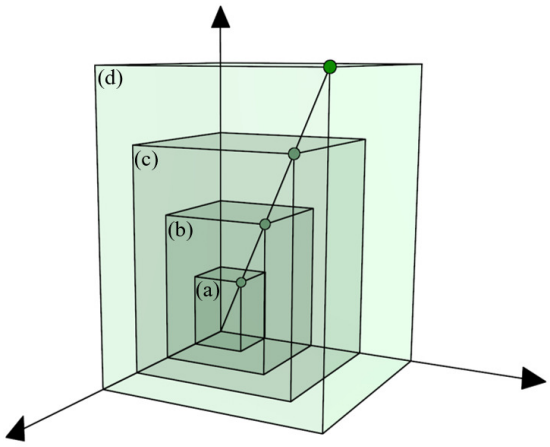


Figura 3.9: Ilustração de possíveis retângulos gerados a partir de um único vetor normal (pontos verdes) variando a distância *r* a partir da origem. a) vetor normal com *r=0,5*; b) vetor normal com *r=1,0*; c) vetor normal com *r=1,5*; d) vetor normal com *r=2,0*.

Pontos chaves

Os pontos chaves são pontos que identificam regiões de interesse em uma imagem e são utilizados para descrever regiões com base em um descritor. Para isto, foi utilizado o algoritmo ORB para encontrar os pontos chaves na imagem D_{Caract}. Dentre os algoritmos de correspondência entre imagens presentes na literatura, como o SIFT (LOWE, 2004) e o SURF (BAY *et al.*, 2008), o ORB apresenta uma melhor relação entre rapidez e eficiência durante a sua execução (KRIG, 2014), sendo propício para aplicações com rastreamento.

Dentre os parâmetros de um ponto chave, o índice de confiança expressa o seu grau de importância. Valores elevados descrevem melhor as suas regiões, entretanto, são mais difíceis de serem encontrados. Portanto, foram selecionados apenas índices maiores que um determinado limiar (0,070). Este valor foi alcançado após realizar diversos testes verificando a quantidade de pontos extraídos ao variar o índice de confiança. Assim, índices a partir deste limiar são mais difíceis de serem encontrados.

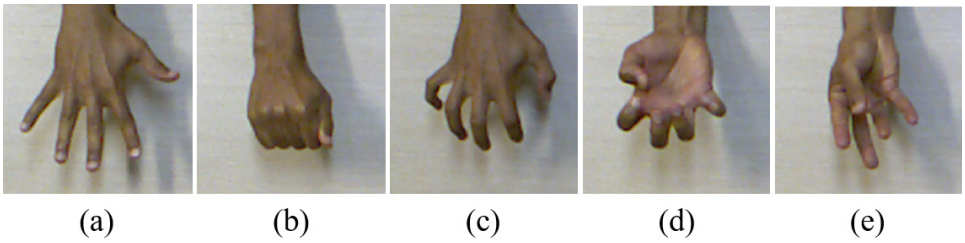
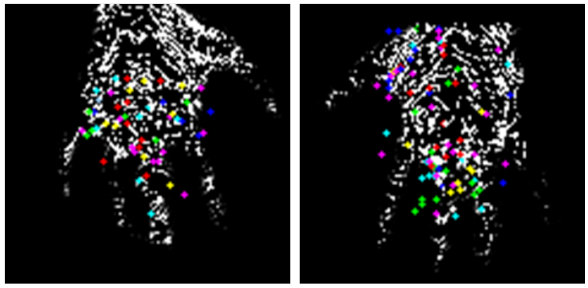


Figura 3.10: Imagens RGB das poses da mão utilizada para a extração das características. a) mão aberta; b) mão completamente fechada; c) mão com as falanges distais e mediais parcialmente retraídas; d) mão invertida com as falanges distais e mediais retraídas; e) mão invertida com dedos estirados para frente.



(a) (b)

Figura 3.11: Imagens da mão (pose aberta) após o processo de extração de características (*(a)*: mão esquerda e *(b)*: mão direita) utilizadas para visualização dos pontos chaves (pontos coloridos) extraídos das 5000 imagens, cada cor representa uma determinada pose.

Com o intuito de extrair o maior número de pontos candidatos com índice igual ou superior a *0,070*[3], foi utilizado um conjunto de 5000 imagens da mão (direita e esquerda) em cinco poses diferentes (FIGURA 3.10). Para cada pose, foram capturadas 500 imagens de forma contínua. Ao todo foram encontrados 173 pontos (FIGURA 3.11) e foram armazenados externamente para serem utilizados apenas no processo de rastreamento. Todo este procedimento de extração de características demanda um elevado custo computacional, sendo realizado de forma independente e à parte da sequência de execução geral. Portanto, o desempenho do sistema está condicionado apenas à extração dos pontos, criação e correspondência entre os descritores de ambas imagens.

3.2.3 Rastreamento da mão

Características locais de uma imagem são identificadas como pontos chaves e seus respectivos descritores (BARFIELD, 2015). Cada conjunto de pontos contém um conjunto de descritores para descrever a região ao seu redor. Tais descritores são utilizados para realizar o processo de correspondência (detecção) entre as imagens.

O processo de criação da imagem de característica (D_{Caract}) demanda um elevado custo computacional, não sendo aplicável ao rastreamento em tempo real. Porém, a imagem D_{BTDE}, utilizada para atribuir valores arbitrários a r (Seção 3.2.2-Pontos chaves), apresenta uma taxa muito baixa de falso positivo durante o processo de correspondência entre as imagens. Além disso, o processo de implementação desta última demanda baixo esforço computacional, sendo suficiente para a aplicação do rastreador.

Para melhorar a extração de pontos sobre a imagem D_{BTDE} foi necessário realizar dois procedimentos: aplicar um filtro de média, utilizando o kernel (3×3) sobre esta; e

[3]Detalhes da definição dos limiares dos índices de confiança e escala estão descritos na seção 4.1.

realizar uma pré-seleção dos pontos chaves extraídos quanto ao seus respectivos índice de confiança e escala.

Conforme descrito anteriormente, os índices de confiança com valores elevados são mais difíceis de serem extraídos, gerando poucos pontos por imagem. No entanto, o processo de extração dos pontos chaves sobre a D_{BTDE} ocorre em tempo de execução, sendo necessário extrair uma elevada quantidade de pontos para a criação dos descritores. Assim, este procedimento é diferente da extração de pontos realizada na imagem modelo, onde há um acúmulo de pontos extraídos em um conjunto de imagens previamente coletadas. Portanto, o limiar utilizado para selecionar os pontos chaves foi inferior ao utilizado na imagem modelo, sendo definido observando a quantidade de pontos correspondidos durante a comparação entre o descritor formado a partir destes pontos e o descritor formado com os pontos da imagem modelo. Valores elevados produziram baixa quantidade de pontos chaves e consequentemente baixa quantidade de pontos correspondidos. Então, valores acima de *0,0001*[4] proporcionam uma quantidade elevada de pontos correspondidos além de descartar pontos que podem gerar falsas correspondências. Adicionalmente, foi observado que ao selecionar pontos chaves que tenham uma escala igual ou superior ao valor *110* diminui o número de falsas correspondências.

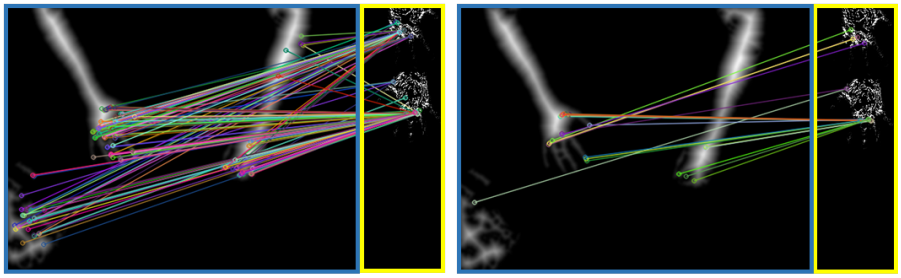
Após a extração e pré-seleção dos pontos chaves em ambas imagens (D_{BTDE} e a imagem modelo), foram criados os seus respectivos descritores. Estes, foram gerados utilizando o algoritmo ORB. Apesar da pré-seleção dos pontos chaves, as correspondências entre eles apresentaram um elevado índice de falso positivo devido à alta variação de similaridade entre as características, sendo necessário um procedimento adicional para tratá-las.

Consenso entre Amostras Aleatórias (CAA)

Para reduzir a quantidade de correspondências falsas ou de baixa importância (FIGURA 3.12-a), resultantes do processo de comparação entre os descritores da imagem modelo e a imagem D_{BTDE}, foi utilizado o método de CAA proposto por Fischler e Bolles (1981). Este, analisa um conjunto de dados e estima uma relação global entre eles, removendo os dados que são divergentes a este. Assim, o total de pontos correspondidos entre as imagens foi reduzido eliminando grande parte das correspondências falsas e/ou as que não correspondem ao mesmo local na imagem modelo (FIGURA 3.12-b).

Os pontos resultantes após a aplicação do CAA apresentam maior probabilidade de acerto durante a correspondência. Assim, grupos de pontos isolados em uma região da imagem são candidatos à presença da mão na imagem.

[4]Detalhes da definição dos limiares dos índices de confiança e escala estão descritos na seção 4.1.



(a) (b)

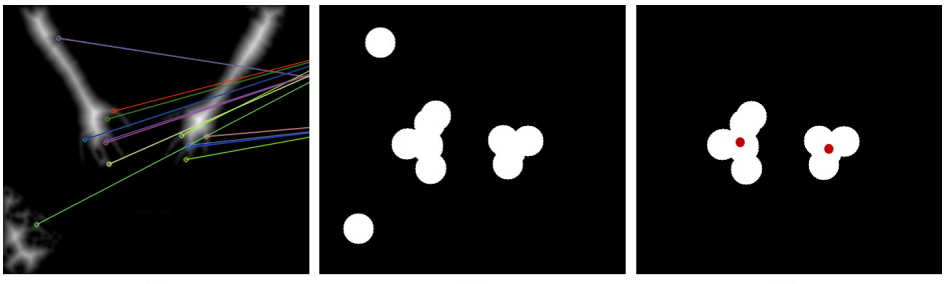
Figura 3.12: Aplicação do método de CAA entre os pontos correspondidos. A borda azul e amarela são as imagens D_{BTDE} e modelo, respectivamente. a) conjunto total de pontos correspondidos com elevado número de falsos positivos; b) conjunto de pontos correspondidos após à aplicação do método de CAA.

Detecção por bolhas

Para verificar a presença da mão e recuperar a sua posição, foi implementado um método baseado na análise de bolhas, sendo estas definidas como o agrupamento de círculos desenhados sobre a posição de cada ponto correspondido sobre a imagem D_{BTDE} após a aplicação do método de CAA (FIGURA 3.13-b). Considerando $Bolhas_{[i]}$ um conjunto de bolhas, $P_{[i]}$ o número de pontos por bolha e $A_{[i]}$ a área quadrada desta, e i o índice de cada bolha, o agrupamento de pontos que indicam a presença da mão é definido pela equação 3.15:

$$Bolhas_{[i]} = \begin{cases} Verdade & \text{para } (P_{[i]} \geq 5) \quad e \quad (2500 \leq A_{[i]} \leq 22500) \\ Falso & \text{para outro caso} \end{cases} \tag{3.15}$$

Os pontos brancos isolados são eliminados por não atenderem ao critério da equação 3.15. Então, o centro da área formada pela bolha indica o centro da mão (FIGURA 3.13-c).



(a) (b) (c)

Figura 3.13: Ilustração do processo de localização da mão por bolhas. a) imagem dos pontos correspondidos após aplicação do CAA; b) círculos brancos na posição dos pontos correspondidos: pontos agrupados formam bolhas, enquanto que pontos isolados são descartados; c) imagem com o centro de cada bolha selecionada (pontos vermelhos) indicando a posição da mão.

3.2.4 Modelo cinemático

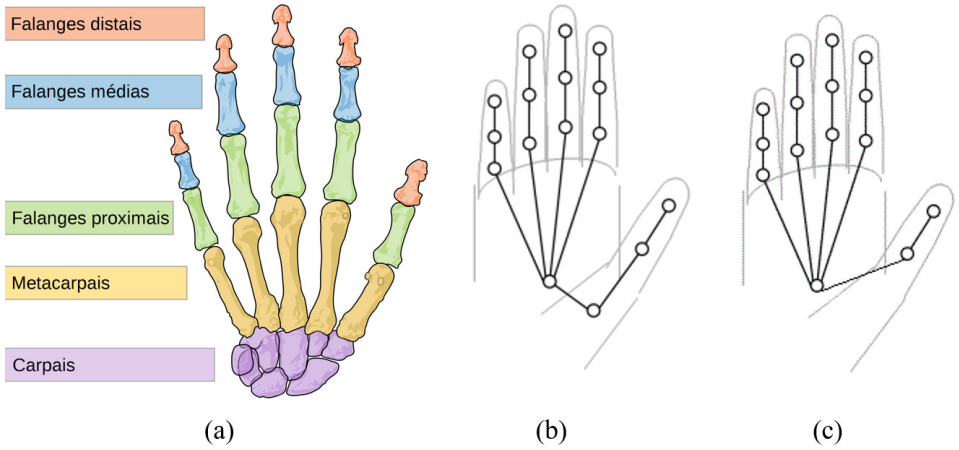


Figura 3.14: Modelo cinemático da mão. a) sistema ósseo da mão (WIKIPÉDIA, 2015); b) modelo cinemático da mão proposto por Sudderth *et al.* (2004); c) modelo cinemático sugerido. Figuras extraídas de Wikipédia (2015) e Sudderth *et al.* (2004), respectivamente.

O modelo cinemático da mão foi proposto com o intuito de estimar o movimento realizado pelos dedos, representando suas articulações internas (ver FIGURA 3.14-a) por meio de ligações de pontos e retas. Porém, o resultado final do método proposto detecta apenas a configuração inicial da mão, sendo necessário aplicar, posteriormente a esta pesquisa, um método para um contínuo rastreio dos dedos. Existem variações na construção da cinemática da mão quanto ao grau de liberdade tratado. O modelo proposto nesta pesquisa é semelhante ao proposto por Sudderth *et al.* (2004) (26 graus de liberdade) como visto na figura 3.14-b e não considerando o metacarpo do polegar (ver FIGURA 3.14-c).

O centro da mão identificado na Seção 3.2.3 (Detecção por bolhas) é utilizado para definir a região de interesse na imagem S_{bin} (equação 3.3) com uma dimensão de 120×120 pixels. Esta região contém apenas a imagem da mão (ver FIGURA 3.15-a) e será utilizada durante a extração do modelo cinemático, sendo identificada como S_{Cine}.

Afinamento

A fim de obter uma forma similar ao sistema ósseo da mão, foi utilizada a técnica de afinamento sugerida por Zhang e Suen (1984). Para diminuir a complexidade desta operação foram aplicados os seguintes procedimentos: S_{TDE} - imagem gerada a partir da aplicação da TDE sobre S_{Cine} (ver FIGURA 3.15-b); S_{TDEG} - imagem gerada a partir da aplicação de um filtro de desfoque Gaussiano (3×3) pixels sobre S_{TDE}; S_{Aguca} - imagem da soma ponderada entre as imagens S_{TDE} e S_{TDEG} (ver FIGURA 3.15-c) utilizando a

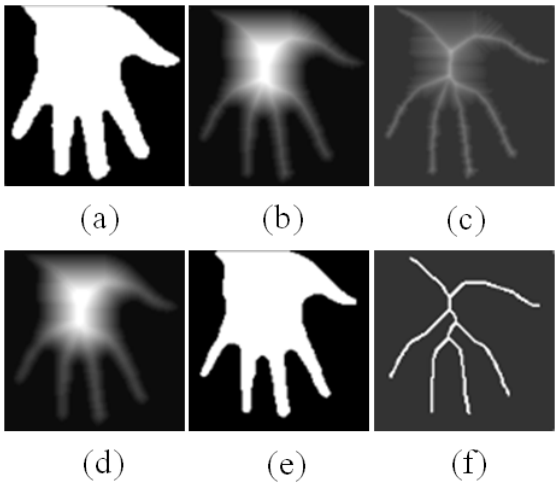


Figura 3.15: Pré-processamento sobre a mão antes de aplicar a técnica de afinamento. a) S_{Cine}: região de interesse (*120×120*) pixels extraída da imagem S_{Bin}; b) S_{TDE}: imagem da mão após aplicar a TDE sobre S_{Cine}; c) S_{Aguca}: imagem após aplicação do filtro de aguçamento *(3 × 3)* pixels sobre *(b)*; d) Imagem após a aplicação do filtro mediano *(5 × 5)* pixels sobre *(c)*; e) Imagem final após a segmentação; f) S_{Afin}: imagem resultante da aplicação da técnica de afinamento sobre *(e)*.

equação 3.16 com os seguintes valores: $\alpha = 1.4$, $\beta = -1.23$ e $\gamma = 0.0$. Esta operação corresponde a aplicação de um filtro de aguçamento sobre a imagem S_{TDE}.

$$S_{Aguca}(x,y) = S_{TDE}(x,y) \times \alpha + S_{Gauss}(x,y) \times \beta + \gamma \tag{3.16}$$

O filtro de desfoque mediano (5×5) foi aplicado sobre S_{Aguca} para destacar as regiões uniformes e remover possíveis ruídos (ver FIGURA 3.15-d). Posteriormente, foi aplicada uma segmentação utilizando a equação 3.17 para eliminar partes das regiões que são irrelevantes à geometria do sistema ósseo da mão (ver FIGURA 3.15-e), sendo esta denominada S_{Seg} . Por fim, a técnica de afinamento foi aplicada sobre esta última gerando a imagem S_{Afin}, conforme a figura 3.15-f.

$$S_{Seg}(x,y) = \begin{cases} 255 & \text{para } S_{Aguca}(x,y) \geq 10 \\ 0 & \text{para outro caso} \end{cases} \tag{3.17}$$

Técnicas morfológicas como a erosão (FISHER *et al.*, 1996) poderiam ser aplicadas diretamente, no entanto, podem comprometer a geometria global do objeto. No entanto, o método descrito acima denigrem apenas as regiões referentes às pontas dos dedos, as quais são recuperadas conforme descrito na subseção 3.2.4-(Ponta dos dedos).

Ramificações

A imagem resultante após o processo de afinamento (S_{Afin}) é semelhante ao esqueleto da mão, cujo os dedos são representados por ramificações. Estas iniciam em uma posição próxima ao centro da mão se estendendo às extremidades dos dedos representando a sua forma e orientação em uma única linha.

Um contorno de um objeto é definido pela forma de sua borda. No entanto, a técnica de afinamento (em sua maioria) representa a sua estrutura e subpartes com apenas linhas, podendo gerar posições duplicadas após aplicação do contorno utilizando o método sugerido por Suzuki e be (1985). O critério utilizado para identificar o início e o fim destes é baseado na análise de pontos consecutivos. Analisando três pontos consecutivos, $P_{Anterior}$, P_{Atual}, $P_{Posterior}$, sobre um contorno C extraídos da imagem (S_{Afin}), a ponta do dedo pode ser identificada quando $P_{Anterior}$ e $P_{Posterior}$ forem iguais e quando forem diferentes, logo após a identificação da ponta do dedo, indicará a base do dedo. É necessário que os pixels do afinamento tenham no máximo dois vizinhos para garantir que existam posições duplicadas ao longo do contorno.

Após a identificação das ramificações (ver FIGURA 3.16-b) são selecionadas apenas as cinco maiores eliminando as restantes (ver FIGURA 3.16-c). Imagens que produzam um número de ramificações inferior a cinco são desconsideradas.

O algoritmo utilizado para extrair as ramificações referente a cada dedo é apresentado abaixo:

Algorithm 1 Algoritmo para extrair ramificações do contorno

```
1:  procedure RAMIFICAÇÃO(C)
2:      P_Atual ← 1
3:      enquanto C faça
4:          P_Anterior ← P_Atual − 1   e   P_Posterior ← P_Atual + 1
5:          enquanto P_Anterior = P_Posterior faça
6:              R ← C[P_Atual]
7:              P_Atual ← P_Posterior
8:              P_Anterior ← P_Anterior − 1
9:              P_Posterior ← P_Posterior + 1
10:         fim enquanto
11:         P_Atual ← P_Atual + 1
12:     fim enquanto
13:     return R
14: fim procedure
```

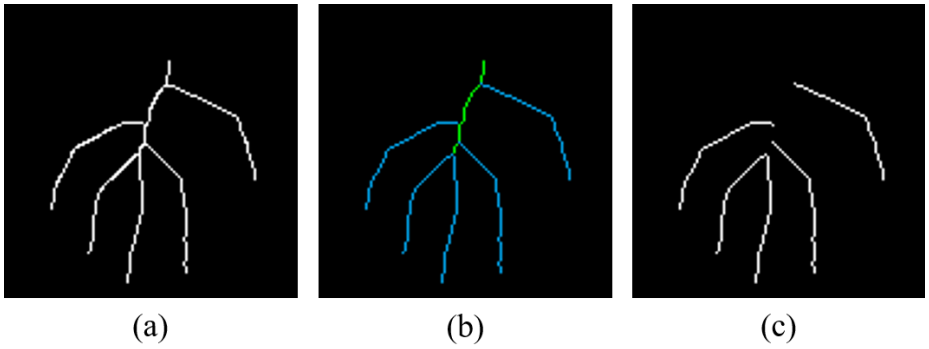


Figura 3.16: Extração das ramificações. a) S_{Afin}: imagem do afinamento da mão; b) identificação das ramificações referentes aos dedos (*azul*) e ao centro da mão (*verde*); c) seleção das cinco maiores ramificações.

Ponta dos dedos

O processo de afinamento visa reduzir a estrutura global de um objeto de acordo com a sua forma, entretanto é inevitável o deterioramento de determinadas regiões. Dessa forma, a extremidade de uma ramificação não corresponde a posição da ponta do dedo na imagem da mão, sendo necessário corrigi-las para obter o tamanho real do dedo. Dado um ponto P_e situado na extremidade da ramificação (início), e um ponto P_{e+c} situado ao centro desta ramificação, k a distância do novo ponto em relação a P_e, e seja *Inclina* o grau de inclinação entre os pontos P_e e P_{e+c}, as pontas dos dedos podem ser localizadas utilizando a abordagem apresentada por Newman e Bull (1997).

$$Inclina = \frac{P_{e+c}(y) - P_e(y)}{P_{e+c}(x) - P_e(x)}\,, \tag{3.18}$$

e

$$y = Inclina \times x + b\,, \tag{3.19}$$

$$P_{Dedo}{}^{5} = \begin{cases} x' = P_{e+c}(x) \pm k \\ y' = k \times Inclina \pm P_{e+c}(y). \end{cases} \tag{3.20}$$

Após este procedimento, foi traçada uma linha entre a extremidade da ramificação e o novo ponto. Esta apresenta um tamanho maior que o dedo original (para *k=15*), sendo necessário eliminar as regiões excedentes ao compará-las com a imagem da mão. A figura 3.17 ilustra todo o procedimento realizados para encontrar os pontos referentes à ponta de cada dedo a partir da extremidade das ramificações.

[5]Os sinais variam de acordo com a posição dos pontos entre si. Positiva quando $P_e(x) > P_{e+c}(x)$ e negativa caso contrário.

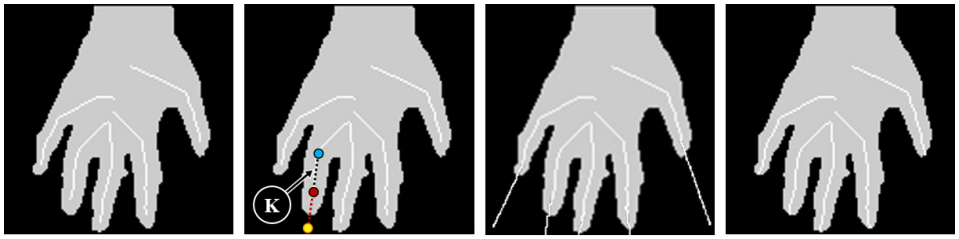


Figura 3.17: Correção das ramificações nas pontas dos dedos. a) ilustração do efeito da deterioração causado pelo processo de afinamento nas extremidades dos dedos; b) demonstração do cálculo do novo ponto: *vermelho* - início da ramificação; *azul* - meio da ramificação; *k* - distância entre o início e o meio da ramificação; *amarelo* - posição aproximada do novo ponto. c) linhas excedentes aos dedos indicam as linhas traçadas entre o início das ramificações e o novo ponto; d) ramificações após a correção.

Eixos medianos

Os eixos dos dedos constituem a parte das ramificações pertencentes apenas ao corpo do dedo representando sua forma bidimensional em uma estrutura unidimensional, sendo então definido como eixos medianos (STAPLES, 1995). Os tamanhos destes eixos, por sua vez, são definidos como a distância entre a base do dedo e à sua extremidade. Assim, para definir o ponto da ramificação referente à base de cada dedo, foi criada uma elipse concêntrica à mão adotando a sua dimensão e orientação.

Para calcular a orientação da mão foi realizada a Análise de Componentes Principais (DUNTEMAN, 1989) sobre os pontos referentes à ramificação do dedo médio, uma vez que este tem maior representação da orientação da mão. Esta abordagem consiste em calcular o arcotangente entre a posição y e x do primeiro elemento dos autovetores extraídos (BRADSKI, 2015). O tamanho do eixo principal e secundário da elipse foi definida como uma dimensão de *40* e *30* pixels, respectivamente, considerando o tamanho da imagem *120* x *120* pixels. As intersecções entre a elipse e as ramificações indicam o ponto de corte, sendo utilizado para definir o início de cada eixo mediano (base do dedo). A figura 3.18-a ilustra a sobreposição da elipse sobre as ramificações, bem como os eixos medianos (FIGURA 3.18-b).

Localização das falanges

Seguindo o modelo cinemático sugerido por Sudderth *et al.* (2004), cada falange é representada por pontos cuja posição indica o ponto mediano do seu tamanho. Este, por sua vez, é determinado utilizando as proporções dos segmentos da mão. A tabela abaixo exibe as proporções de cada segmento em relação ao tamanho do seu respectivo dedo (BURYANOV; KOTIUK, 2010).

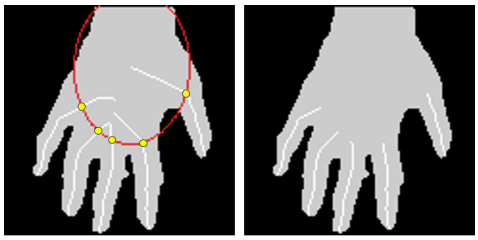


Figura 3.18: Extração dos eixos medianos. a) sobreposição da elipse (vermelho) sobre a mão e suas intersecções com as ramificações referente ao ponto de corte (pontos amarelos); b) eixos medianos correspondentes a cada dedo.

Seja E um vetor contendo as posições dos pontos de um eixo mediano, T o seu tamanho, e P_{Fal} a proporção de uma determinada falange (P_D (distal), P_M (medial) e P_P (proximal)), a posição do centro desta é determinado pela equação 3.21.

$$Centro_{Fal} = E\left[\left(\frac{T \times P_{Fal}}{2}\right) + Indice_{Fal}\right] \tag{3.21}$$

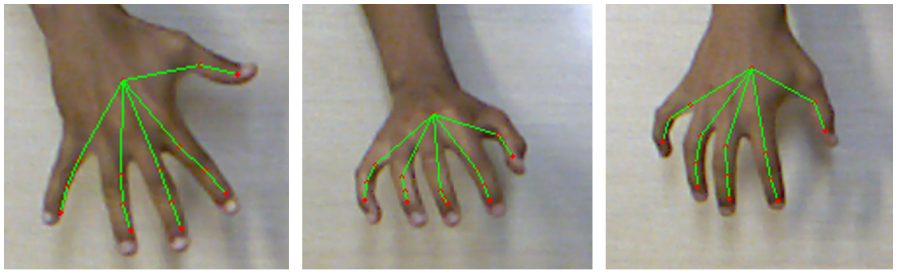
O $Indice_{Fal}$ presente na equação 3.22 determina o início de cada falange e é fundamental para localizar o seu centro. Ao final, os centros das falanges juntamente com o centro da mão formam a estrutura cinemática, sendo representada por uma estrutura interligada por linhas e pontos.

$$Indice_{Fal} = \begin{cases} Indice_D = 0 \\ Indice_M = (P_D * T) \\ Indice_P = (P_D * T) + (P_M * T) \end{cases} \tag{3.22}$$

A construção deste modelo cinemático está condicionado a pose da mão (aberta) e ao rastreamento no estágio anterior. Contudo, é possível estimar as falanges em diferentes poses quando são identificadas as cinco ramificações da mão. No entanto, a integridade do modelo pode ser comprometida quando o eixo mediano não corresponder ao tamanho do dedo devido a mudança do ponto de vista (FIGURA 3.19-b,c).

Índice dos dedos	I	II	III	IV	V
Ápice (*ponta*) + falange distal	49,36	27,80	26,32	28,25	33,92
Falange médias	-*	30,97	33,11	33,61	31,99
Falange proximal	50,64	15,52	15,33	18,49	24,72

Tabela 3.1: Relação das proporções de cada falange em relação ao tamanho do dedo, desde o polegar (I) até o dedo mínimo (V). A proporção da falange distal é calculada considerando o tamanho da ponta do dedo; (*) O polegar não contém a falange medial.



(a) (b) (c)

Figura 3.19: Estrutura final do modelo cinemático da mão. As linhas verdes representam as ligações entre o centro de cada falange (pontos vermelhos) e o centro da mão. a) sobreposição do modelo cinemático construído sobre a mão (RGB). b) e c) modelo cinemático identificado em diferentes poses.

3.3 Considerações finais

Neste capítulo, foram descritas detalhadamente as abordagens utilizadas na construção do método proposto, bem como soluções para atender demandas surgidas durante a pesquisa. Foi desenvolvido um método para detectar bordas (ADV-4) baseado na diferença entre os pixels vizinhos, possibilitando segmentá-las por limiarização. Foi proposto um método para extração de características (VNOR) sobre o mapa de profundidade utilizando a orientação do vetor normal de cada pixel. Para isto, utilizou-se a informação da TDE sobre a imagem da mão para exceder a descrição das características. Este método pode ser aprimorado para extrair determinados tipos de características em um mapa de profundidade por possibilitar o uso de variadas informações no cálculo do volume da normal.

Utilizando o método ORB foi possível extrair os pontos chaves e seus respectivos descritores da imagem capturada e da imagem modelo da mão para realizar o processo de correspondência entre ambas imagens. Para diminuir o número de falsas correspondência foi utilizado o método de CAA. Por fim, para recuperar a posição do centro da mão com base nos pontos correspondidos foi proposto um método baseado na detecção por bolhas.

O modelo cinemático construído visa apenas identificar as posições das falanges dos dedos enquanto a mão estiver aberta e paralela ao plano de imagem do sensor, servindo apenas como parte inicial para um trabalho posterior a esta pesquisa.

Capítulo 4

Experimentos e Resultados

Conteúdo

Este capítulo descreve os experimentos realizados para definir os limiares que foram utilizados para selecionar os pontos chaves extraídos da imagem modelo e da imagem da cena. Além disso, é realizada a análise dos resultados alcançados nesta pesquisa.

4.1 Experimentos

Esta seção é destinada a demonstração do procedimento utilizado para selecionar valores de parâmetros utilizados nos métodos das seções 3.2.2 e 3.2.3. Os testes realizados nesta seção são destinados a definição dos limiares utilizados para seleção dos pontos chaves da imagem modelo e da imagem da cena.

Limiar 1

O limiar utilizado para selecionar os pontos chaves extraídos da imagem modelo foi proposto visando selecioná-los com base no seu índice de confiança e assim reduzir a taxa de falso positivo durante o processo de correspondência entre os descritores das

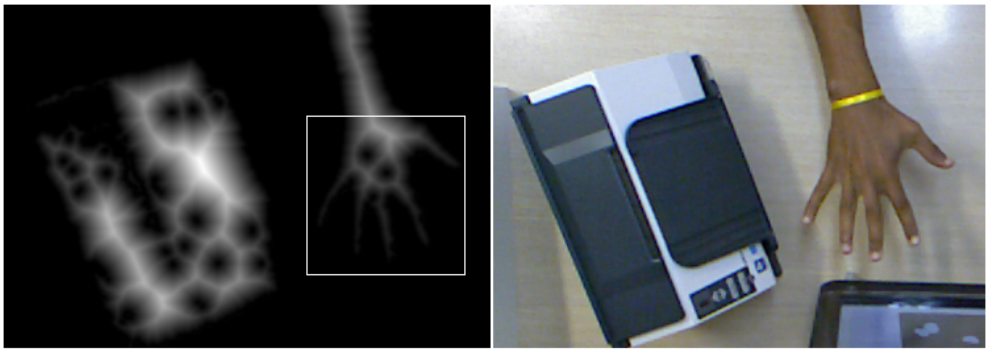


Figura 4.1: Ilustração do cenário utilizado para análise dos valores durante a definição dos limiares. a) imagem RGB da cena com a mão e um objeto com formas planas (scanner) à sua esquerda, o qual foi utilizado para induzir a falsos positivos durante o processo de correspondência; b) imagem em profundidade processada pela TDE, onde o quadrado é posto para limitar a região da mão.

imagens. Os pontos chaves com índice de confiança elevado são mais difíceis de serem extraídos, porém, produzem melhores resultados durante a correspondência.

Para a realização dos experimentos foi feita a comparação entre os descritores da imagem modelo e da imagem da cena após a extração dos pontos chaves de ambas imagens. Entretanto, a cena utilizada foi composta da mão aberta e de um objeto com características similares à mão, para induzir a falsos positivos durante o processo de correspondências. Assim, é possível avaliar a quantidade de pontos correspondidos corretamente ao variar o índice de confiança dos pontos chaves extraídos da imagem modelo. Para identificar os pontos correspondidos pertencentes à mão foi posto um quadrado delimitador para indicar a região da mão. Assim, todos os pontos correspondidos que estiverem dentro deste são considerados verdadeiros positivos e todos que estiverem fora são considerados falsos positivos (ver FIGURA 4.1).

Durante a extração dos pontos chaves da imagem modelo foram selecionados apenas os pontos chaves que tivesse o índice de confiança acima de um determinado limiar. Para identificar o melhor limiar, este procedimento foi realizado quatro vezes com diferentes limiares, analisando a precisão dos pontos correspondidos. Os pontos chaves foram extraídos três vezes para cada limiar, sendo utilizada sua média. Assim, foi escolhido o limiar que apresentou a maior taxa de precisão.

Observando os valores utilizados e a taxa de precisão alcançada (ver FIGURA 4.2), o valor 0,007 alcançou uma taxa de precisão igual a 13.93%, sendo este o limiar escolhido para seleção dos pontos chaves. Foi constatado que a quantidade de pontos chaves com índices de confiança acima de 0,007 é quase nula para a quantidade de imagens utilizadas.

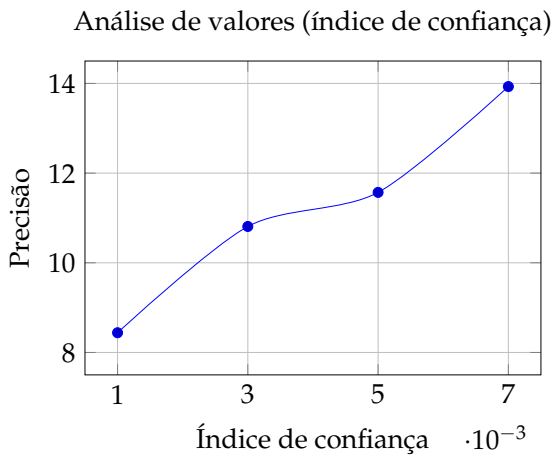


Figura 4.2: Análise de valores para selecionar os pontos chaves da imagem modelo de acordo com o índice de confiança. Relação entre valores dos índices de confiança testados durante a seleção dos pontos (eixo x) e suas respectivas taxas de precisão alcançadas (eixo y).

Limiar 2

Após a realização das correspondências entre as imagens utilizando o limiar 0,007, foi realizado o mesmo procedimento para selecionar os pontos chaves da imagem da cena. Ao contrário da imagem modelo, onde os pontos chaves são extraídos de um conjunto de imagens previamente capturada, os pontos chaves da imagem da cena são extraídos a partir de uma única imagem por vez. Assim, a quantidade de pontos é menor e seus respectivos índices de confiança apresentam, em sua maioria, baixos valores.

Ao todo foram realizadas seis correspondências entre os pontos da imagem modelo (extraídos com o limiar 0,007) e os pontos da imagem da cena. Para cada correspondência foi utilizado um valor diferente para selecionar estes de acordo com seus respectivos índices de confiança. Foi utilizada a mesma cena empregada na definição do limiar 1 juntamente com a precisão para avaliar a taxa de pontos correspondidos corretamente em cada correspondência. Para cada valor utilizado para selecionar os pontos chaves da imagem da cena, o processo de correspondências foi realizado três vezes, sendo a taxa de precisão a média destas três.

Os resultados da análise demonstrou que os melhores valores para selecionar os pontos chaves foram 0,00020, 0,00010 e 0,00015 gerando uma precisão de 17.11%, 16,88% e 16,86%, respectivamente (ver FIGURA 4.3). Embora o 0,00020 tenha alcançado a maior taxa de precisão é possível notar que a precisão diminui a partir do valor 0,00010 como é observado no valor 0,00025, sendo evento caracterizado como uma exceção. Portanto o valor selecionado para o limiar foi igual a 0,00010. A quantidade de pontos chaves com índice de confiança acima de 0,00025 é quase nula, portanto o limiar foi definido com o valor 0,00010.

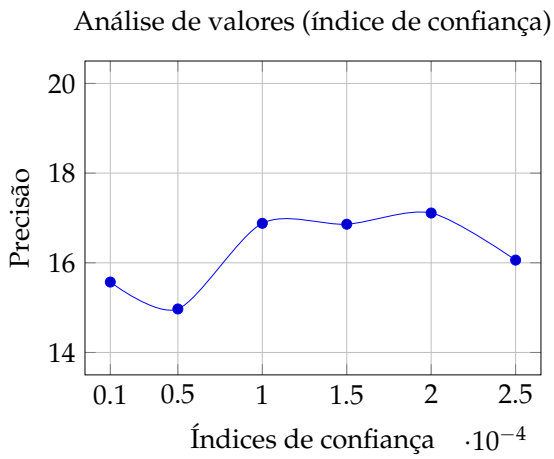


Figura 4.3: Análise de valores para selecionar os pontos chaves da imagem da cena de acordo com os seus respectivos índice de confiança. Relação entre valores dos índices de confiança testados durante a seleção dos pontos (eixo x) e suas respectivas taxas de precisão alcançadas (eixo y).

A quantidade de pontos chaves extraídos da imagem da cena é muito inferior a quantidade extraída na imagem modelo, pois são realizados sobre apenas uma imagem, assim a quantidade de pontos chaves com índice de confiança elevados é baixa, tornando inviável a utilização de limiares elevados. Desta maneira, foi utilizada a escala de cada ponto chave para complementar a seleção destes pontos.

Para selecionar os pontos chaves de acordo com suas respectivas escalas, foi realizado o mesmo procedimento para a definição dos limiares anteriores. Os pontos chaves de ambas imagens foram extraídas, porém foram utilizados os limiares definido anteriormente 0,007 e 0,0001, respectivamente. Em seguida, foram realizadas seis correspondências entre estes pontos. Para cada uma dessas os pontos chaves da imagem da cena foram pré-selecionados de acordo com uma determinada escala e em seguida avaliado a precisão alcançada. Cada correspondência foi realizada três vezes sobre o mesmo valor, sendo a precisão final a média das taxas de precisão resultantes das três correspondências.

Os resultados demonstraram que a escala 110 alcançou a maior taxa de precisão, sendo que não ocorreram pontos chaves com uma escala maior que esse valor, como é observado com o valor 120 (ver FIGURA 4.4). Assim os limiares utilizados para selecionar os pontos chaves da imagem da cena foram baseados no índice de confiança e escala de cada ponto chaves com os seguintes valores 0,0001 e 110, respectivamente.

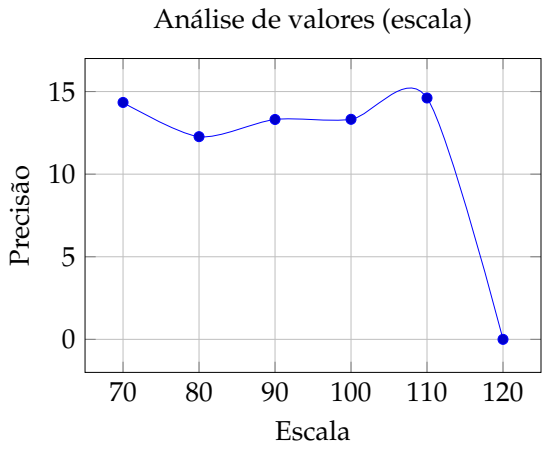


Figura 4.4: Análise de valores para selecionar os pontos chaves da imagem da cena de acordo com suas respectivas escalas. Relação entre valores das escalas testados durante a seleção dos pontos (eixo x) e suas respectivas taxas de precisão alcançadas (eixo y).

4.2 Análise dos resultados

Esta seção descreve uma série de análises feitas sobre os resultados obtidos durante a aplicação do rastreamento proposto (detecções sucessivas da mão). Toda a análise é baseada nos pontos correspondidos entre a imagem capturada e o conjunto de imagens modelo da mão. São utilizados os seguintes critérios: taxa de detecção e a análise do desempenho computacional, as quais são descritas nas seções seguintes.

4.2.1 Taxa de detecção

A detecção da mão na imagem capturada ocorre quando há presença de pontos correspondidos, pois os testes realizados nesta seção contêm apenas a presença da mão na cena. Assim, a taxa de detecção é definida pela proporção do número de imagens detectadas sobre o total de imagens capturadas, sendo considerado uma detecção ideal uma taxa igual a *1*.

Para realização dos testes foram capturadas 500 imagens para cada pose, sendo estas realizadas por 6 pessoas diferentes. Assim, para cada cenário testado, a taxa de detecção foi aplicada de duas maneiras diferentes: de forma sequencial e por grupo, explicadas nas subseções seguintes.

Taxa de detecção sequencial

Este procedimento calcula a média das taxas de detecção de as pessoas, as quais são determinadas pela média das taxas de detecção de cada pose realizada. Esta taxa

tem como intuito verificar a robustez do método em relação ao impacto causado pelo fenômeno de flutuação dos pixels ao longo do tempo, onde um valor de um pixel apresenta variações ao longo de uma sequência de imagens capturada de uma cena estática (ANDERSEN *et al.*, 2012). Além da taxa, é calculada a média de imagens detectadas continuamente, ou seja, o número médio de imagens sucessivas em que ocorreu a detecção da mão.

Taxa de detecção por grupo

Esta taxa foi proposta para avaliar o desempenho do detector quanto à diversidade de mãos, onde imagens de uma mesma pose, de diferentes pessoas, são utilizadas para calcular a taxa de detecção. Ao contrário da taxa de detecção sequencial, que faz comparação entre imagens de uma mesma pose por pessoa, a taxa de detecção por grupo compara as imagens de uma pose de uma pessoa com as imagens, da mesma pose, referentes a outras pessoas. Assim, seu cálculo considera o número de imagens de uma mesma pose, entre as pessoas, que houve detecção sobre o número total de pessoas, resultando em uma taxa de detecção para cada imagem de uma pose, sendo necessário calcular a média para cada pose.

Cenário 1

Este cenário calcula as taxas de detecção sequencial e por grupo para o conjunto de poses utilizadas na extração dos pontos chaves da imagem modelo (Seção 3.1), sendo estas realizadas por 6 pessoas diferentes e compostas por 500 imagens cada uma.

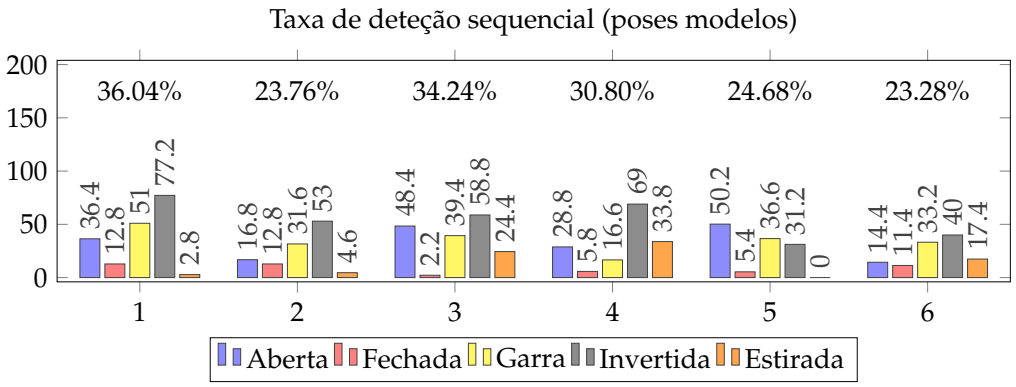


Figura 4.5: Análise da taxa de detecção sequencial sobre as poses modelo. Cada cor indica a taxa de detecção para uma determinada pose realizada por uma pessoa (legenda). Ao todo foram utilizadas seis pessoas diferentes (1 à 6), sendo o valor ao topo referente a taxa de detecção de cada pessoa. A taxa de detecção sequencial é calculada a partir da média de todas as taxas de detecção por pessoa.

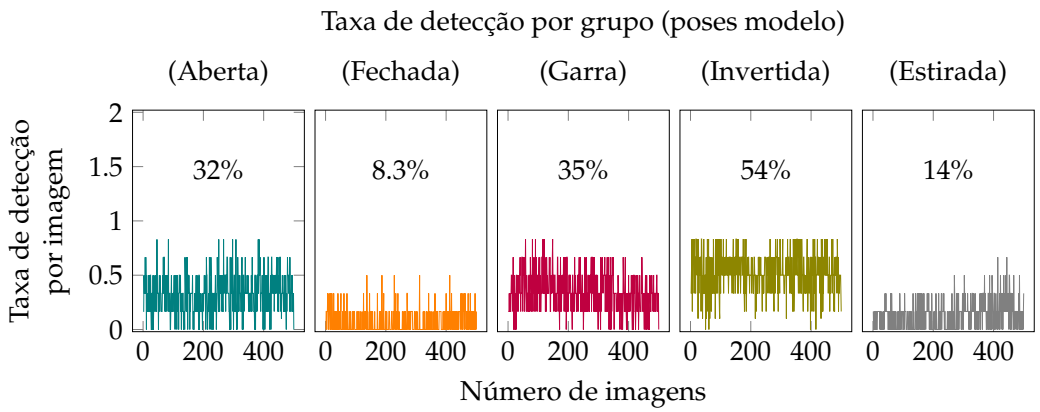


Figura 4.6: Análise da taxa de detecção por grupo sobre as imagem das poses modelo (nomes acima dos gráficos). A taxa de detecção (eixo y) é calculada considerando as imagens de uma mesma pose realizadas por diferentes pessoas, onde cada pose é composta por 500 imagens (eixo x). Em seguida, é realizada a média para cada pose (topo), sendo a taxa de detecção por grupo a média de todas as taxas de detecção em cada pose.

Os testes realizados apresentaram uma taxa de detecção sequencial igual a 28,8% com um desvio padrão de 5,64%. As poses *Invertida* e *Fechada* apresentaram a maior e a menor taxa de detecção comum à todos os indivíduos tendo uma média igual a 54,87% e 8,40%, respectivamente. O número de pontos correspondidos, por pose, variou entre as pessoas, sendo diretamente relacionada à forma da mão destas, sendo a maior e a menor taxa de detecção por pessoa iguais a 36,04% e 23,28%, respectivamente (FIGURA 4.5). Está análise demonstra que o detector apresentou um baixo desempenho quanto ao efeito de flutuação dos pixels sobre o conjunto de imagens das poses modelo.

A taxa média de detecção por grupo foi igual a 44% com um desvio padrão igual a 26%, sendo a maior e a menor taxa iguais a 83% e 14% referentes as poses *Fechada* e *Estirada*, respectivamente (FIGURA 4.6). Este resultado mostra que o detector apresentou baixa robustez quanto a diversidade das mãos ao realizar as poses modelo. Portanto, a taxa de detecção final é definida como a média da taxa de detecção sequencial e por grupo sendo igual a 36,4%. Estes resultados indicam que é necessário melhorar o processo de detecção quanto a determinados tipos de poses, bem como torná-lo mais robusto quanto ao efeito de flutuação dos pixels e diversidade das mãos.

Cenário 2

Esta análise tem por intuito verificar o comportamento do detector quanto à presença de poses diferentes ao conjunto das poses modelo. Para isso, foi realizada a mesma análise do cenário 1, porém com poses diferentes (ver FIGURA 4.12). Estas poses foram realizadas pelas mesmas 6 pessoas do cenário 1, sendo também compostas por 500 imagens cada uma.

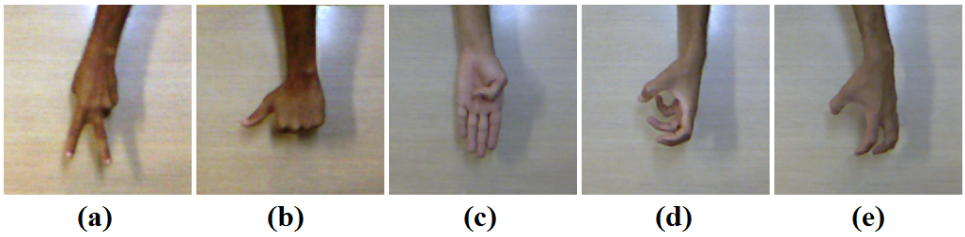


Figura 4.7: Imagens das poses utilizadas nos testes do cenário 2. a) mão fechada com os dedos indicador e médio estirados (*V**); b) mão completamente fechada com o pelegar estirado (*Positivo*); c) mão aberta com os dedos juntos e polegar completamente retraído (*B**); d) mão aberta com os dedos levemente retraídos e posicionada de perfil (C*); e) mão em forma de garra posicionada de perfil (*Garra Perfil*). *(*)* - letras da Linguagem Brasileira de Sinais (SILVEIRA, 2016).

A taxa média de detecção sequencial foi igual a 30,3% com um desvio padrão de 12.6%. As poses *B* e *V* apresentaram a maior e a menor taxa de detecção média entre todas as pessoas com valores iguais a 81,7% e 8,8%, respectivamente. O número de pontos correspondidos foi muito inferior ao apresentado no cenário 1, sendo a maior e menor taxa de detecção por pessoa igual a 47,40% e 40,32%, respectivamente (ver FIGURA 4.8). Estas análises demonstram uma ruim adaptação do detector em assimilar poses diferentes das poses modelo. O desempenho do detector quanto ao efeito de flutuação dos pixels, sobre este conjunto de poses, também foi inferior ao apresentado no cenário 1.

Portanto, a taxa de detecção média por grupo apresentou 46% e desvio padrão igual a 36%, semelhante ao apresentado no cenário 1, porém com um desvio padrão mais elevado. As poses *B* e *Positivo* apresentaram a maior e a menor taxas com valores iguais a 82% e 13%, respectivamente (ver FIGURA 4.9).

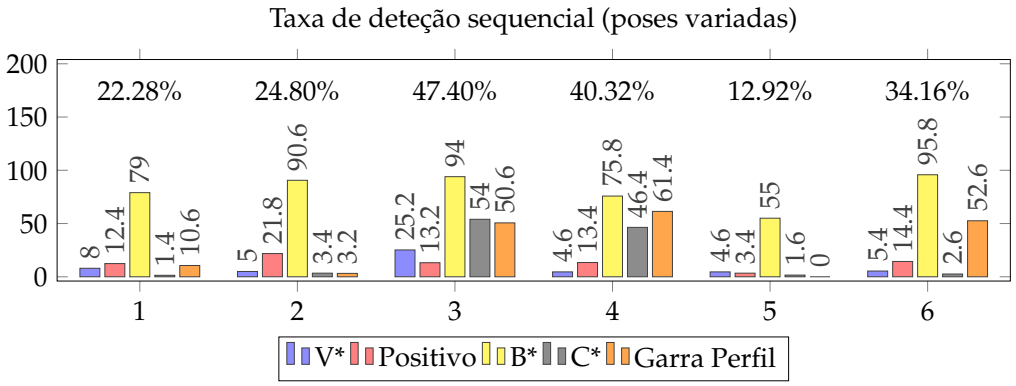


Figura 4.8: Análise da taxa de detecção sequencial sobre poses variadas. Cada cor indica a taxa de detecção para uma determinada pose realizada por uma pessoa (legenda). Ao todo foram utilizadas seis pessoas diferentes (1 à 6), sendo o valor ao topo referente a taxa de detecção de cada pessoa. A taxa de detecção sequencial é calculada a partir da média de todas as taxas de detecção por pessoa. (*) - letras da Linguagem Brasileira de Sinais (SILVEIRA, 2016).

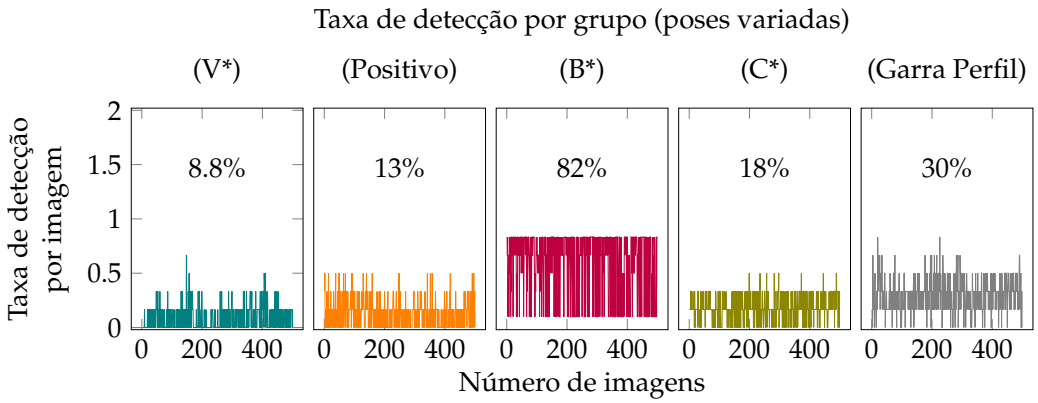


Figura 4.9: Análise da taxa de detecção por grupo sobre as imagens de poses variadas (nomes acima dos gráficos). A taxa de detecção (eixo y) é calculada considerando as imagens de uma mesma pose realizadas por diferentes pessoas, onde cada pose é composta por 500 imagens (eixo x). Em seguida, é realizada a média para cada pose (valores dentro dos gráficos), sendo a taxa de detecção por grupo a média de todas as taxas de detecção de cada pose. (*) - letras da Linguagem Brasileira de Sinais (SILVEIRA, 2016).

A taxa de detecção final foi igual a 38,15%. Este valor foi superior à taxa de detecção sobre as poses do treinamento, isto indica a grande quantidade de características presentes nas poses utilizadas. Entretanto, esta análise confirma a baixa robustez do detector quanto ao reconhecimento das mãos sob a presença de diferentes pessoas.

Cenário 3

Este cenário visa avaliar o detector quanto a sua aplicação para o reconhecimento de gestos. Foram propostos quatro gestos realizados a partir do conjunto das poses modelos contendo apenas 40 imagens cada um.

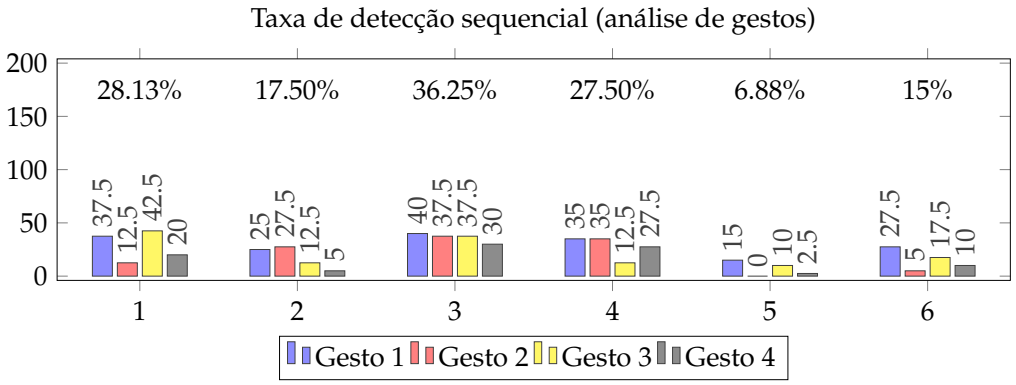


Figura 4.10: Análise da taxa de detecção sequencial sobre os gestos realizados a partir do conjunto de poses modelo. Cada cor indica a taxa de detecção para um determinado gesto realizado por uma pessoa (legenda). Ao todo foram utilizadas seis pessoas (1 a 6), sendo o valor dentro de cada gráfico a taxa de detecção por pessoa. A taxa de detecção sequencial é calculada a partir da média de todas as taxas de detecção por pessoa.

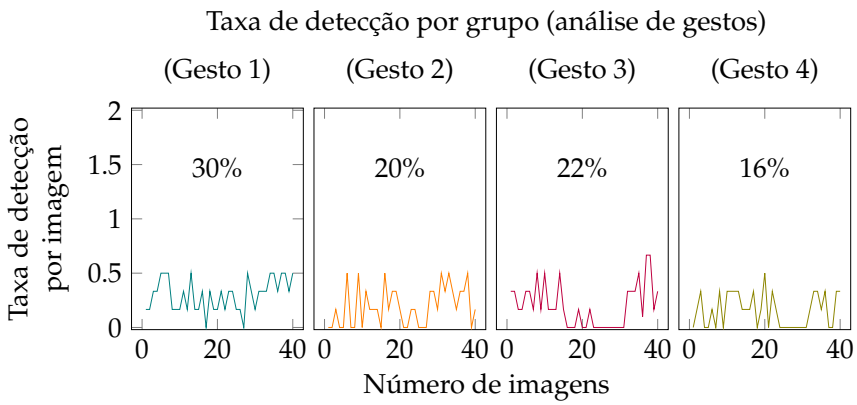


Figura 4.11: Análise da taxa de detecção por grupo sobre as imagens de gestos realizados a partir do conjunto das poses modelo (nomes acima dos gráficos). A taxa de detecção (eixo y) é calculada considerando as imagens de um mesmo gestos realizados por diferentes pessoas, onde cada gesto é composto por 40 imagens (eixo x). Em seguida, é realizada a média para cada gesto (valores dentro dos gráficos), sendo a taxa de detecção por grupo a média de todas as taxas de detecção de cada gesto.

A taxa de detecção sequencial foi igual a 21,88% com um desvio padrão de 10,67%. A maior e menor média de imagens detectadas por gestos foi é igual a 30% e 15% respectivamente, sendo referentes aos gestos 1 e 4 (ver FIGURA 4.10).

A taxa de detecção por grupo alcançou uma valor igual a 22% e o desvio padrão igual a 6%, similarmente à taxa de detecção sequencial, os gestos 1 e 4 apresentaram a maior e a menor média com valores iguais a 30% e 16% (ver FIGURA 4.11).

Está análise demonstra o baixo desempenho do método quanto a análise gestos, pois a txa de detecção final foi igual a 21,94%. Além disso, é necessário torná-lo mais robusto quanto a variabilidade das mãos.

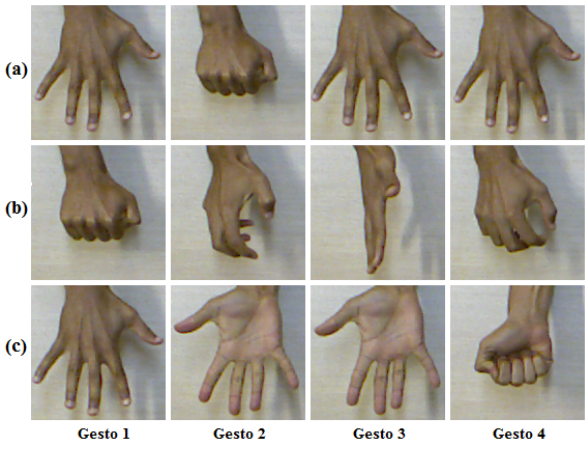


Figura 4.12: Imagens dos gestos utilizados para verificar a aplicação do detector quanto ao reconhecimento de gestos. Ao todo foram realizados quatro gestos (gesto 1 ao gesto 4) contendo 40 quadros cada um, onde as letras *a*, *b* e *c* indicam o início (quadro 1), meio (quadro 20) e fim (quadro 40) de cada gesto.

Cenário 4

Este cenário foi proposto visando avaliar o detector quanto a presença de objetos diversos na cena. Para realizar esta análise foram utilizadas três cenas diferentes, cada qual com diferentes objetos. Neste caso não houve a realização de poses ou gestos, apenas a presença da mão aberta entre os objetos na cena. Ao todo foram utilizadas cinco pessoas sendo coletadas 500 imagens por cena.

O processo de correspondência visa detectar pontos pertencentes apenas à mão, ou seja, estes pontos são considerados verdadeiro positivo quando há correspondência. Porém, podem surgir pontos correspondentes pertencentes a outros objetos, neste caso foram considerados como falsos positivos. Assim, foi utilizado a métrica de precisão para avaliar os resultados.

Os pontos correspondidos pertencentes a mão abrangem podem ocorrer em diferentes regiões desta, assim, todos os pontos correspondidos contidos no contorno da mão são considerados verdadeiros positivos e todos os pontos pertencentes ao contorno de objetos são considerados falsos positivos, contornos brancos e vermelhos na FIGURA 4.13, respectivamente.

Para esta análise o detector apresentou uma precisão média igual a 14,72% com um desvio padrão igual a 3,82%. A cena em que apresentou a maior e o menor precisão foram as cenas 3 e 2 com médias iguais a 24,84% e 9,22%, respectivamente (ver FIGURA 4.14). Estes resultados indicam que cenas com objetos de características planas acentuadas apresentam uma elevada taxa de falso positivos, enquanto que cenas com apenas objetos de forma irregular (mochila) apresentam baixa taxa de falsos positivos. Este fato ocorre devido a semelhança plana que a mão apresenta.

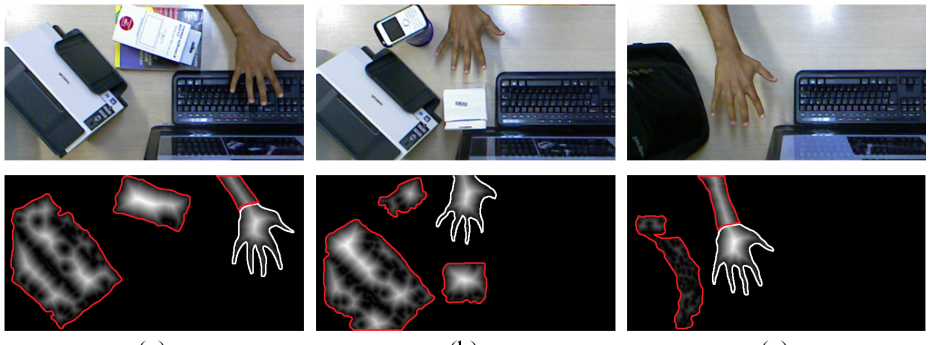


Figura 4.13: Detecção da mão com a presença de objetos na cena: Topo - imagem RGB da cena. Rodapé - imagem da TDE segmentada. a) (cena 1) e b) (cena 2) apresentam a mão com presença de objetos que contém partes planas (caixa, livro, scanner) e partes do braço; c) (cena 3) apresenta objetos com forma irregular (mochila). Contornos vermelhos indicam área que não deve ocorrer detecção (falso positivo), enquanto brancas indicam áreas de interesse (verdadeiro positivo).

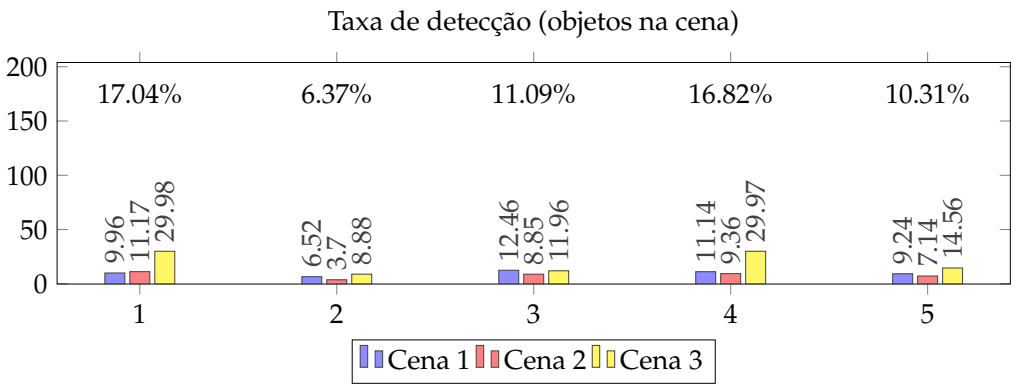


Figura 4.14: Análise da precisão da mão em cenas com a presença de diferentes objetos. Cada cor indica uma determinada cena (legenda). Ao todo foram utilizadas cinco pessoas diferentes (1 a 5), sendo o valor dentro de cada gráfico a precisão média de todas as cenas para cada pessoa. O valor da precisão final é calculado a partir da média da precisão encontrada em cada pessoa.

Ao fim da análise do detector nestes cenários é necessário melhorar seu poder discriminativo para detecção de diferentes poses, para a diversidade de mãos utilizadas e para a detecção em cenários com a presença de objetos com forma similar à mão.

4.2.2 Desempenho computacional

Esta etapa analisa o desempenho computacional durante a extração de características e a correspondência de descritores das duas imagens. O procedimento da primeira análise apresenta um nível de complexidade maior e consequentemente possui uma eficiência muito inferior a segunda análise.

Utilizando um computador Intel Core i5 com 3 Gigabytes de RAM juntamente com o OpenNI (PRIMESENSE, 2016), as imagens foram capturadas à uma taxa de 30 quadros por segundo (qps). Durante o processo de extração de característica a taxa média de desempenho foi de 2,9 quadros por segundo, enquanto o processo de rastreamento apresentou uma taxa média de 7,9 qps.

Avaliação Final

Utilizando a métrica de exatidão, o trabalho apresentado por Tang *et al.* (2015) utilizaram uma abordagem baseada em *Deep Belief Networks* (DBN) para reconhecer as posturas da mão e alcançou uma taxa mínima de 96.78% de exatidão para a detecção de poses do conjunto de treino, tendo uma velocidade de 12 quadros por segundo. Paralelamente, foram conduzidos testes comparativos utilizando o HGO junto com a MVS, onde o menor resultado para a detecção de poses do conjunto de treinamento foi igual a 81,12% a uma velocidade de 1,6 quadros por segundo. A abordagem apresentada

Métodos	DBN	(HGO+MVS)	KTSL	(VNOR+ORB)
Maior taxa de detecção	98,97%	92,65%	77,17%	54,87%
Menor taxa de detecção	96,78%	81,12%	3,27%	8,40%

Tabela 4.1: Análise de taxas de detecção de diferentes abordagens para detecção das mãos. A tabela exibe os melhores e piores casos de detecção das abordagens DBN, (HGO+MVS) e KTSL apresentados por Tang *et al.* (2015) e Lee *et al.* (2016) respectivamente, juntamente com a abordagem proposta neste trabalho (VNOR+ORB).

por Lee *et al.* (2016) propõe um sistema para reconhecimento de linguagem de sinais denominado Kinect-based Taiwanese Sign-Language (KTSL), onde o melhor resultado para reconhecimento de posturas do conjunto de treinamento é igual a 77,17%. No entanto, a melhor taxa de detecção apresentada pela nossa abordagem para a detecção de poses do conjunto de treino (cenário 1) foi igual a 54.87% a uma velocidade de 7,9 quadros por segundo (ver TABELA 4.1).

Ao final a nossa abordagem apresentou um resultado inferior a todos os métodos comparados, superando apenas o KTSL quanto a taxa de detecção mínima de imagens do conjunto de treino.

4.3 Considerações finais

Há dois pontos fundamentais que poderiam ser alterados para melhorar o método propostos: a escolha da imagem para atribuir os valores de r durante a aplicação do *VNOR* e a metodologia utilizada para realizar a correspondência entre as imagens. Os dados utilizados para atribuir os valores a r é baseado apenas no cálculo da transformada da distância Euclidiana agravando o efeito de flutuação dos pixels, uma vez que esta utiliza informação da borda da mão, a qual apresenta grande instabilidade. Outros tipos de informações poderiam ser explorados para aumentar o poder discriminativo do detector (texturas, cor, bordas, entre outras).

Foi necessário reduzir o conjunto de características para executar as correspondências entre os descritores utilizando o algoritmo ORB, uma vez que este é otimizado para aplicações em tempo real, porém, esta redução diminuiu poder discriminativo do detector. Como alternativa poderia ser utilizado classificadores como o MVS para realizar uma detecção mais precisa da mão.

Outras poses modelos poderiam ser exploradas para extrair características que a melhor descreva, possibilitando uma detecção mais robusta.

Capítulo 5

Considerações finais

O sistema proposto apresentou uma abordagem para detectar e rastrear as mãos utilizando dados de profundidade adquiridos do sensor Kinect. Adicionalmente, foi proposta uma abordagem para extração do modelo cinemático desta. Durante a fase de pré-processamento foi desenvolvido um algoritmo para detectar bordas baseado no acúmulo da diferença dos vizinhos mais próximos (ADV-4), sendo necessário realizar análises com outras abordagens da literatura para avaliar a sua aplicação. O procedimento de extração de característica proposto calcula o volume do vetor normal no mapa de profundidade (VNOR) possibilitando adicionar outras informações (cor, textura, contorno, entre outras) para aumentar seu poder discriminativo. Embora tenha sido utilizado para realizar correspondências entre imagens, poderá ser empregado em outras áreas da visão computacional que envolva o reconhecimento de padrões quando há informação de profundidade, podendo ser utilizada com técnicas de aprendizado de máquina (classificadores) para alcançar resultados mais promissores.

O rastreamento é baseado na correspondência entre descritores da imagem capturada com um conjunto de imagens modelo da mão e não é realizado de forma temporal, mas quadro a quadro. O método ORB foi utilizado para realizar esta comparação devido principalmente ao seu desempenho em relação a outras abordagens existentes. Entretanto, tem como principal desvantagem a redução do conjunto de características para sua aplicação, podendo afetar o poder discriminativo da abordagem. O uso da técnica de correspondência entre imagens possibilitou realizar o estágio de extração de características à parte do rastreamento sem influenciar no desempenho geral do sistema.

O detector alcançou uma taxa de detecção final igual a 27% com um desvio padrão de 9,6%, sendo executado a uma velocidade de 7,9 quadros por segundo. Embora tenha apresentado uma taxa de detecção média de 29,5% e desvio padrão de 9,16% sobre os testes realizados no cenário 1 e 2, o método conseguiu detectar poses do conjunto de

treino e poses variadas, ambas realizadas por diferentes pessoas. Com uma taxa de detecção média inferior a 30%, os testes realizados no cenário 3 indicam a inviabilidade do método para aplicações em reconhecimento de gestos. A principal deficiência do método é evidenciada nos testes do cenário 4, pois apresenta elevadas taxas de falsos positivos em cenas com objetos de forma plana, sendo necessário aumentar o poder discriminativo do detector. Como possível solução, a abordagem de comparação entre descritores poderá ser substituída pela utilização de classificadores (MVS), assim, todas as características extraídas poderão ser aproveitadas.

A descrição proposta da cinemática da mão visa criar um modelo a partir das falanges dos dedos da mão enquanto estiver aberta (dedos completamente estirados) e paralela ao plano de imagem do sensor, sendo necessário implementar o rastreamento contínuo dos dedos, em um estágio complementar a esta pesquisa. Este procedimento poderá ser realizado utilizando a abordagem apresentada por Sudderth *et al.* (2004), porém, utilizando dados de profundidade, por exemplo.

Alguns pontos poderiam ser explorados para melhorar os resultados alcançados: verificar quais informações poderão melhor descrever a mão para atribuir valores ao tamanho da normal durante o cálculo do seu volume; diminuir o efeito de flutuações dos pixels presentes no mapa de profundidade; explorar outras abordagens mais eficientes para o cálculo do vetor normal dos pixels.

Referências

ABREU, J. *Reconstrução da cinemática da mão em pacientes com hanseníase*. 2013. 87 p. Monografia (Graduação em Engenharia Mecânica) — UFRJ (Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2013.

ANDERSEN, M. R. *et al. Kinect Depth Sensor Evaluation for Computer Vision Applications*. Dinamarca: Aarhus University, Department of Engineering, 2012. 37 p.

BARFIELD, W. *Fundamentals of Wearable Computers and Augmented Reality*. 2. ed. Boca Raton, FL, EUA: CRC Press, 2015. 739 p.

BAY, H. *et al.* Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, v. 110, n. 3, p. 346–359, jun. 2008.

BERGH, M. V. d.; GOOL, L. V. Combining rgb and tof cameras for real-time 3d hand gesture interaction. In: IEEE WORKSHOP ON APPLICATIONS OF COMPUTER VISION (WACV), 2011, Kona, HI, EUA. EUA: IEEE, 2011. p. 66–72.

BILLINGHURST, M.; BUXTON, B. *Gesture Based Interaction 14.1 Capítulo 14: GESTURE BASED INTERACTION*. 2016. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.304.7919>. Acesso em: 21 jan. 2016.

BORGEFORS, G. Distance transformations in digital images. *Computer Vision, Graphics, and Image Processing*, v. 34, n. 3, p. 344–371, jun. 1986.

BOZGEYIKLI, G.; BOZGEYIKLI, E.; ISLER, V. Introducing tangible objects into motion controlled gameplay using microsoft kinect tm. *Computer Animation and Virtual Worlds*, Reino Unido, v. 24, n. 3-4, p. 429–441, aug. 2013.

BRADSKI, G. Open computer vision library. *Dr. Dobb's Journal of Software Tools*, nov. 2000.

BRADSKI, G. *Open Computer Vision Library: Introduction to Principal Component Analysis (PCA)*. 2015. Disponível em: <http://docs.opencv.org/master/d1/dee/tutorial_introduction_to_pca.html#gsc.tab=0>. Acesso em: 28 nov. 2015.

BURYANOV, A.; KOTIUK, V. Proportions of hand segments. *International Journal of Morphology*, Chile, v. 28, n. 3, p. 755–758, set. 2010.

CALONDER, M. *et al.* Brief: Binary robust independent elementary features. In: EUROPEAN CONFERENCE ON COMPUTER VISION, 11., 2010, Heraclião, Creta, Grécia. Berlim, Heidelberg: Springer-Verlag, 2010. p. 778–792.

CAMPOS, G. *Sistema para fisioterapia baseado na plataforma Kinect*. 2013. 70 p. Dissertação (Mestrado Integrado em Engenharia Eletrotécnica e de Computadores) — Universidade do Porto, Porto-Portugal, 2013.

CHEN, N. *et al.* Human-aided robotic grasping. In: IEEE INTERNATIONAL SYMPOSIUM ON ROBOT AND HUMAN INTERACTIVE COMMUNICATION, 21., 2012, Paris, França. EUA: IEEE, 2012. p. 75–80.

CORDELLA, F. *et al.* Patient performance evaluation using kinect and monte carlo-based finger tracking. In: IEEE RAS EMBS INTERNATIONAL CONFERENCE ON BIOMEDICAL ROBOTICS AND BIOMECHATRONICS (BIOROB), 4., 2012, Roma, Itália. EUA: IEEE, 2012. p. 1967–1972.

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 2005, San Diego, CA. EUA: IEEE Computer Society, 2005. p. 886–893.

DIJKSTRA, E. W. A note on two problems in connexion with graphs. *NUMERISCHE MATHEMATIK*, v. 1, n. 1, p. 269–271, dec. 1959.

DUNTEMAN, G. H. *Principal Components Analysis*. Newbury Park, CA, EUA: SAGE Publications, 1989. 96 p.

FENG, Z. *et al.* Real-time fingertip tracking and detection using kinect depth sensor for a new writing-in-the air system. In: INTERNATIONAL CONFERENCE ON INTERNET MULTIMEDIA COMPUTING AND SERVICE, 4., 2012, Wuhan, China. Nova Iorque, NY, EUA: ACM, 2012. p. 70–74.

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

FISHER, R. *et al. HIPR - The Hypermedia Image Processing Reference*. Estados Unidos: J. Wiley & Sons, 1996. 24 p.

FLEISCH, D. *A Student's Guide to Maxwell's Equations*. Nova Iorque, NY, EUA: Cambridge University Press, 2008. 134 p.

GALLAGHER, G. *Hand Detection Demo*. 2013. Disponível em: <http://wiki.ros.org/mit-ros-pkg/KinectDemos/HandDetection>. Acesso em: 28 nov. 2013.

GIRALDO, D. R. *et al.* Kernel based hand gesture recognition using kinect sensor. In: IMAGE, SIGNAL PROCESSING, AND ARTIFICIAL VISION (STSIVA), 17., 2012, Antioquia, Colombia. EUA: IEEE, 2012. p. 158–161.

HONGYONG, T.; YOULING, Y. Finger tracking and gesture recognition with kinect. In: INTERNATIONAL CONFERENCE ON COMPUTER AND INFORMATION TECHNOLOGY (CIT), 12., 2012, Chengdu, Sichuan, China. Los Alamitos, CA, USA: IEEE, 2012. p. 214–218.

JAIN, R.; KASTURI, R.; SCHUNCK, B. G. *Machine Vision*. Nova Iorque, NY, EUA: McGraw-Hill, Inc., 1995. 549 p.

JIU, M. *et al.* Human body part estimation from depth images via spatially-constrained deep learning. *Pattern Recognition Letters*, v. 50, p. 122–129, dec. 2014.

KEAN, S.; HALL, J.; PERRY, P. *Meet the Kinect: An Introduction to Programming Natural User Interfaces*. 1. ed. Berkeley, CA: Apress, 2011. 220 p.

KHOSHELHAM, K.; ELBERINK, S. O. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, Suíça, v. 12, n. 2, p. 1437–1454, fev. 2012.

KIM, A. R.; RHEE, S. Y. Robot control by using multi-modes. In: INTERNATIONAL CONFERENCE ON SOFT COMPUTING AND INTELLIGENT SYSTEMS (SCIS) AND INTERNATIONAL SYMPOSIUM ON ADVANCED INTELLIGENT SYSTEMS (ISIS), 6., 13., 2012, Cobe, Japão. EUA: IEEE, 2012. p. 2051–2052.

KREJOV, P.; BOWDEN, R. Multi-touchless: Real-time fingertip detection and tracking using geodesic maxima. In: IEEE INTERNATIONAL CONFERENCE AND WORKSHOPS ON AUTOMATIC FACE AND GESTURE RECOGNITION (FG), 10., 2013, Xangai, China. EUA: IEEE, 2013. p. 1–7.

KRIG, S. *Computer Vision Metrics: Survey, Taxonomy, and Analysis*. Berkeley, CA: Apress, 2014. 465 p.

KRIM, H.; HAMZA, A. B. *Geometric Methods in Signal and Image Analysis*. Cambridge, Inglaterra: Cambridge University Press, 2015. 295 p.

LECUN, Y.; HUANG, F. J.; BOTTOU, L. Learning methods for generic object recognition with invariance to pose and lighting. In: PROCEEDINGS OF THE 2004 IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, 2004, Washington, DC, EUA. EUA: IEEE, 2004. p. II–97–104 Vol.2.

LEE, G. C.; YEH, F.; HSIAO, Y. Kinect-based taiwanese sign-language recognition system. *Multimedia Tools Appl.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 75, n. 1, p. 261–279, jan. 2016.

LEE, J.; KUNII, T. L. Model-based analysis of hand posture. *IEEE Comput. Graph. Appl.*, v. 15, n. 5, p. 77–86, set. 1995.

LI, H. *et al.* Hands detection based on statistical learning. In: FIFTH INTERNATIONAL SYMPOSIUM ON COMPUTATIONAL INTELLIGENCE AND DESIGN (ISCID), 5., 2012, Hangzhou, China. EUA: IEEE, 2012. p. 227–230.

LI, Y. Hand gesture recognition using kinect. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING AND SERVICE SCIENCE (ICSESS), 3., 2012, Pequim, China. EUA: IEEE, 2012. p. 196–199.

LIANG, H.; YUAN, J.; THALMANN, D. 3d fingertip and palm tracking in depth image sequences. In: ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA, 20., 2012, Nara, Japan. New York, NY, USA: ACM, 2012. p. 785–788.

LOWE, D. G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, v. 60, n. 2, p. 91–110, nov. 2004.

MILES, R. *Start Here! Learn the Kinect API*. 1. ed. Estados Unidos: Microsoft Press, 2012. 272 p.

MINNEN, D.; ZAFRULLA, Z. Towards robust cross-user hand tracking and shape recognition. In: INTERNATIONAL CONFERENCE ON COMPUTER VISION WORKSHOPS (ICCV WORKSHOPS), 13., 2011, Barcelona, Espanha. EUA: IEEE, 2011. p. 1235–1241.

NEWMAN, G.; BULL, R. *Essential Skills in Maths*. Cheltenham, Inglaterra: Nelson Thornes, 1997. 184 p.

OGIELA, M.; HACHAJ, T. *Natural User Interfaces in Medical Image Analysis: Cognitive Analysis of Brain and Carotid Artery Images*. 1. ed. Suíça: Springer Publishing Company, 2014. 288 p.

OIKONOMIDIS, I.; KYRIAZIS, N.; ARGYROS, A. Markerless and efficient 26-dof hand pose recovery. In: *Computer Vision – ACCV*. Berlim, Alemanha: Springer, 2010. p. 744–757.

OKA, K.; SATO, Y.; KOIKE, H. Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION, 5., 2002, Washington, DC, EUA. EUA: IEEE, 2002. p. 429–434.

OSADCHY, M. *et al.* Synergistic face detection and pose estimation with energy-based model. *The Journal of Machine Learning Research*, Estados Unidos, v. 8, p. 1017–1024, may. 2005.

OUEDRAOGO, Y.; AOKI, Y. Finger posture estimation using 3d medial axes. In: INTERNATIONAL CONFERENCE ON HUMAN SYSTEM INTERACTIONS (HSI), 7., 2014, Lisboa, Portugal. EUA: IEEE, 2014. p. 71–75.

PARK, S. *et al.* 3d hand tracking using kalman filter in depth space. *EURASIP Journal on Applied Signal Processing*, Alemanha, v. 2012:36, p. 18, dec. 2012.

PHADTARE, L. K.; KUSHALNAGAR, R. S.; CAHILL, N. D. Detecting hand-palm orientation and hand shapes for sign language gesture recognition using 3d images. In: WESTERN NEW YORK IMAGE PROCESSING WORKSHOP (WNYIPW), 15, 2012, Nova York, NY, EUA. EUA: IEEE, 2012. p. 29–32.

PISHARADY, P. K.; SAERBECK, M. Robust gesture detection and recognition using dynamic time warping and multi-class probability estimates. In: IEEE SYMPOSIUM ON COMPUTATIONAL INTELLIGENCE FOR MULTIMEDIA, SIGNAL AND VISION PROCESSING (CIMSIVP), 2013, Singapura. EUA: IEEE, 2013. p. 30–36.

PISSANETZKY, S. *Vectors, Matrices and C++ Code*. Texas, EUA: Sergio Pissanetzky, 2004. 365 p.

PRIMESENSE. *Open Natural Interaction SDK (OPENNI) para Linux*. 2016. Disponível em: <http://openni.ru/index.html>. Acesso em: 21 jan. 2016.

QIAN, C. *et al.* Realtime and robust hand tracking from depth. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 2014, Columbus, OH, EUA. EUA: IEEE, 2014. p. 1106–1113.

RAHEJA, J. L.; CHAUDHARY, A.; SINGAL, K. Tracking of fingertips and centers of palm using kinect. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE, MODELLING AND SIMULATION (CIMSIM), 3., 2011, Langkawi, Malásia. EUA: IEEE, 2011. p. 248–252.

RAUTARAY, S. S.; AGRAWAL, A. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, Estados Unidos, v. 43, n. 1, p. 1–54, jan. 2012.

REN, Z.; YUAN, J.; ZHANG, Z. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In: ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA, 19., 2011, Scottsdale, Arizona, USA. Nova Iorque, NY, EUA: ACM, 2011. p. 1093–1096.

ROSTEN, E.; DRUMMOND, T. Machine learning for high-speed corner detection. In: EUROPEAN CONFERENCE ON COMPUTER VISION, 9., 2006, Graz, Austria. Berlin, Heidelberg: Springer-Verlag, 2006. p. 430–443.

RUBLEE, E. *et al.* Orb: An efficient alternative to sift or surf. In: IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION (ICCV), 13., 2011, Barcelona, Espanha. EUA: IEEE, 2011. p. 2564–2571.

RYAN, D. J. *Finger and gesture recognition with Microsoft Kinect*. 2012. 67 p. Dissertação (Mestrado em Ciência da Computação) — Universidade de Stavanger, Stavanger-Noruega, 2012.

SAMADANI, A.; KULIC, D.; GORBET, R. Multi-constrained inverse kinematics for the human hand. In: ANNUAL INTERNATIONAL CONFERENCE OF THE IEEE ENGINEERING IN MEDICINE AND BIOLOGY SOCIETY (EMBC), 34., 2012, San Diego, CA, EUA. EUA: IEEE, 2012. p. 6780–6784.

SANCHES, J. M.; MICó, L.; CARDOSO, J. *Pattern Recognition and Image Analysis: 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, June 5-7, 2013, Proceedings*. 1. ed. Berlim, Alemanha: Springer, 2013. 900 p.

SANTOS, E. S.; LAMOUNIER, E. A.; CARDOSO, A. Interaction in augmented reality environments using kinect. In: SYMPOSIUM ON VIRTUAL REALITY (SVR), 13., 2011, Uberlândia, Minas Gerais, Brasil. EUA: IEEE, 2011. p. 112–121.

SANTOS, T. N.; OLIVEIRA, L. R. Finger phalanx detection and tracking by contour analysis on rgb-d images. In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES (SIBGRAPI), 28., 2015, Salvador. Porto Alegre: Sociedade Brasileira de Computação, 2015. Disponível em: <http://urlib.net/sid.inpe.br/sibgrapi/2015/07.13.14.19>. Acesso em: 19 Jan. 2016.

SCHRODER, M. *et al.* Real-time hand tracking using synergistic inverse kinematics. In: IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION (ICRA), 2014, Hong Kong, China. EUA: IEEE, 2014. p. 5447–5454.

SILVEIRA, C. H. *Curso à Distância de Especialização em Educação Especial*. 2016. Disponível em: <http://coral.ufsm.br/edu.especial.pos/images/libras.pdf>. Acesso em: 15 fev. 2016.

STAPLES, J. *6th International Symposium, ISAAC '95 Cairns, Australia, December 4 - 6, 1995. Proceedings Proceedings*. 1. ed. Berlim, Alemanha: Springer-Verlarg, 1995. 450 p.

SUAREZ, J.; MURPHY, R. R. Hand gesture recognition with depth images: A review. In: IEEE RO-MAN, 21., 2012, Paris, França. EUA: IEEE, 2012. p. 411–417.

SUDDERTH, E. B. *et al.* Visual hand tracking using nonparametric belief propagation. In: PROCEEDINGS OF THE 2004 CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION WORKSHOP (CVPRW), 2004, Washington, DC, EUA. EUA: IEEE, 2004. p. 189.

SUZUKI, S.; BE, K. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, v. 30, n. 1, p. 32–46, abr. 1985.

TANG, A. *et al.* A real-time hand posture recognition system using deep neural networks. *ACM Trans. Intell. Syst. Technol.*, ACM, New York, NY, USA, v. 6, n. 2, p. 21:1–21:23, mar. 2015.

TANG, C. *et al.* Hand tracking and pose recognition via depth and color information. In: IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND BIOMIMETICS (ROBIO), 2012, Cantão, China. EUA: IEEE, 2012. p. 1104–1109.

TANG, S. *et al.* Histogram of oriented normal vectors for object recognition with a depth sensor. In: ASIAN CONFERENCE ON COMPUTER VISION, 11., 2013, Daejeon, Coreia. Berlim, Heidelberg: Springer-Verlag, 2013. p. 525–538.

TOMPSON, J. *et al.* Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics*, v. 33, n. 5, p. 169:1–169:10, sep. 2014.

VICENTE, A. P.; FAISAL, A. A. Calibration of kinematic body sensor networks: Kinect-based gauging of data gloves "in the wild". In: IEEE INTERNATIONAL CONFERENCE ON BODY SENSOR NETWORKS (BSN), 10., 2013, Cambridge, EUA. EUA: IEEE, 2013. p. 1–6.

WEN, Y. *et al.* A robust method of detecting hand gestures using depth sensors. In: IEEE INTERNATIONAL WORKSHOP ON HAPTIC AUDIO VISUAL ENVIRONMENTS AND GAMES (HAVE), 2012, Munique, Alemanha. EUA: IEEE, 2012. p. 72–77.

WIGDOR, D.; WIXON, D. *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. 1. ed. São Francisco, CA, EUA: Elsevier, 2011. 264 p.

WIKIPÉDIA. *Falange (Ilustração das falanges da mão de um hominídeo)*. 2015. Disponível em: <https://pt.wikipedia.org/wiki/Falange>. Acesso em: 28 nov. 2015.

XU, D. *et al.* Integrated approach of skin-color detection and depth information for hand and face localization. In: IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND BIOMIMETICS (ROBIO), 2011, Karon Beach, Phuket. EUA: IEEE, 2011. p. 952–956.

YANG, C. *et al.* Gesture recognition using depth-based hand tracking for contactless controller application. In: IEEE INTERNATIONAL CONFERENCE ON CONSUMER ELECTRONICS (ICCE), 2., 2012, Berlim, Alemanha. EUA: IEEE, 2012. p. 297–298.

YIN, J. *et al.* Research on real-time object tracking by improved camshift. In: INTERNATIONAL SYMPOSIUM ON COMPUTER NETWORK AND MULTIMEDIA TECHNOLOGY (CNMT), 2009, Wuhan, China. EUA: IEEE, 2009. p. 1–4.

ZHANG, T. Y.; SUEN, C. Y. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, Estados Unidos, v. 27, n. 3, p. 236–239, mar. 1984.

Finger phalanx detection and tracking by contour analysis on RGB-D images

Thalisson Santos
Intelligent Vision Research Lab
Federal University of Bahia
Salvador, Brazil
thalisson.nobre@ufba.br

Luciano Oliveira
Intelligent Vision Research Lab
Federal University of Bahia
Salvador, Brazil
http://www.ivisionlab.eng.ufba.br/

***Abstract*—In this paper we propose a method for identification of the finger phalanges based on the analysis of hand contour in RGB-D sensors. The proposed method is able to partially identify and track the kinematic structure of the fingers. The tracking was performed using the ORB algorithm to match points between a template with some hand images (in different poses) and the image captured. The principal component analysis was performed to compute the hand orientation relative to the image plane. The system will be used as a starting point for a full tracking of the fingers articulated movement.**

***Keywords*-phalanges; hand kinematic; RGB-D cameras;**

I. Introduction

The analysis of the hand kinematics can be useful in many applications that accurate information about the joints of the fingers is necessary, since human machine interaction up to medical diagnosis or treatment. The use of RGB-D cameras has been widely exploited, eliminating the mandatory use of invasive sensors (e.g. data glove) for accurate results. Visual analysis of kinematics in clinical practice often seeks the use of reflective markers on the joints of the hand to facilitate the mapping of points of interest. However, there may be occlusion and displacement during the execution of movements, hindering the efficient diagnosis of patients, especially those with prostheses [1]. The analysis the movement of hand requires high accuracy during the acquisition of the kinematic variables (joint angles), and complex kinematic systems are often used, these being expensive [2]. This project proposes the creation of a method for detection the phalanges of the finger in depth images captured from an RGB-D camera aimed at creating an inexpensive kinematics system for the analysis of hands. Our method is based on the hand contour analysis by to extract the fingers location (phalanges) without the markers. The kinematic model is built from the phalanges and its position is recalculated every image captured resulting in a constant hand detection process. Currently the method performing the procedure when the hand is fully open and parallel to the sensor image plane.

A. Related work

Hand tracking is a challenging task and many works use markers to achieve efficient results [3] and [4]. However this type of approach is intrusive and usually interferes the movement of the hand preventing its application in clinical research [2], for instance. Different techniques have been proposed to track the hand and estimate its movement. The approach proposed by [5] uses a 3D virtual model of the hand using a point cloud to obtain the hand pose by inverse kinematics while [6] uses the convolutional networks to recover poses of the hand. Using the medial axis technique for extracting fingers and phalanges, [7] presents an approach which demonstrates a solution with inverse kinematics to infer parts of occluded fingers. Hand tracking is invariant to rotation only for the camera view axis. Using the concept of inverse kinematics [8] proposed a multi-constrained approach to accurately reproduce the movement of hand joints from motion data previously captured.

II. Outline of the proposed method

The acquisition of images from the RGB-D camera in the scene was strategically set to capture only the upper limb user when the hands are extended on the table and in parallel to lens of sensor.Therefore the lens of the kinect sensor was positioned vertically and slightly set above the computer for a top vision and targeted to hand action. Similar setup can be found in [9].In order to achieve the hand kinematics estimation was necessary to apply a pre-processing on the image to reduce the inherent noise and minimize the variability of pixels presented at the image edges. At the end, the detection of the phalanges is performed by using a template matching approach.

For fingers phalanges identification was utilized techniques of Euclidean Distance Transform (EDT) and thinning algorithm. For hand tracking the ORB algorithm was applied along with the Principal Component Analysis.

The following sections describe in detail all the approach.

A. Preprocessing

This step is necessary to prepare the original image for the following stages.

Be *D* the depth image acquired by an RGB-D camera, P_i the values of the pixels in the image, and L_{bfr} and L_{aft} the thresholds used to the start and the end of the range, respectively. The resulting image (D_{seg}) can be obtained by

$$D(x, y) = P_i \tag{1}$$

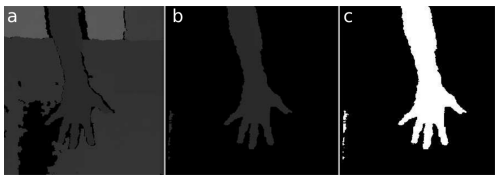


Fig. 1. Segmentation pipeline; (a) Depth map; (b) Segmented depth map; (c) Resulting binary image.

$$D_{seg}(x,y) = \begin{cases} P_i & \text{para } L_{bfr} \leq P_i \leq L_{aft} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$D_{bin}(x,y) = \begin{cases} 255 & \text{for } P_i \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The values closer to 0.5 meters are susceptible to higher incidence of noise and considering the desktop table as background the values determined for L_{bfr} and L_{aft} were 0.8 and 0.95 meters, respectively. The binary image (D_{bin}) and the final segmentation result was extracted from the depth map using the equation 3. This binary image is passed to next step of the proposed method (figure 1c).

The direct use of the contour of segmented hand is not appropriated due the high instability of pixels presented in these areas. In certain cases the pixel intensity causes loss or significant deformation over the hand contour. The following algorithms perform a linear operation on contour points in order to replace them and resulting in a softer effect of the hand contour: the first aims to repair discontinuities while the second applies smoothing.

To apply the following methods is necessary to convert the hand contour to a point vector by using the method proposed in [10] and to reduce the computational cost we simplified the vector calculation with the method proposed in [11].

1) Filling discontinuities: Be *V* an array of points with *n* corresponding positions of contour size; $Pt_i(x_a, y_a)$ and $Pt_{i+k}(X_b, y_b)$ arbitrary points at distance of *K* elements being *i* the index of the first point and *(i + k)* the second. A midpoint *M(x_m, y_m)* can be calculated from the arithmetic mean of their respective coordinates.

$$M(x_m, y_m) = \begin{cases} x_m = & (x_a + x_b)/2 \\ y_m = & (y_a + y_b)/2 \end{cases} \quad (4)$$

Every point in the *V* vector present among the points Pt_i and Pt_{i+k} will have their coordinates recalculated by do a simple arithmetic mean with the coordinate of midpoint obtained. Finally two new points will be chosen and all points that are between them will replaced. This procedure is performed on all the hand contour resulting in a smoothing contour at end of process.

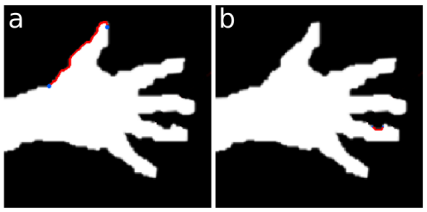


Fig. 2. Setting the *k* constant; (a) Setting a high value for *k* generates a hand contour with more point than necessary; (b) Setting a low value for *k*, generates a hand contour with inadequate amount of points.

The purpose of this method is to repair isolated discontinuities belonging the same contour region. Considering high value for *k* implies long intervals encompassing points from different regions of the object (figure 2a). The inverse can not produce satisfactory results due the low number of points per range (figure 2b). Thus based on test analysis performed on hand contour *k* was set to value equal to eight.

2) Smoothing: This procedure is similar to algorithm introduced previously but the constant *k* is set to two. The goal is replace only the intermediate point of a set of three points. Be *P1*, *P2* and *P3* three consecutive points in hand contour the midpoint *M(xm, ym)* resultant is calculated by using the arithmetic mean and assigned to the intermediate point (*P2*). After apply this algorithm the regions with surpluses tend to be more uniform.

After applying these two last steps was necessary to apply a Gaussian filter to reduce the effect of expansion generated by smoothing process and consequently was necessary convert it back to binary image using the equation 5.

$$D_{BinPr}(x,y) = \begin{cases} 255 & \text{for } P_i > 245 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

For performance purposes the vector was again simplified. The figure 3 illustrates an overview of the steps of the preprocessing stage.

B. Detection of the phalanges

The detection stage is based on template matching technique and checks the presence of the hand in certain regions of image.

Two fixed regions (R_{lef} and R_{rgh}) of dimension 120x120 pixels were defined on the resulting binary image (D_{BinPr}) to verify the presence of the hand (left and right respectively).

The method consists of applying a logical operation between the pixels of two images: the hand model I_{obj} and the target hand I_{targ}. The hand model is composed by two images extracted from the detection region (R_{lef} and R_{rgh}) on the D_{BinPr} in each frame captured.The target is a fixed model and was previously extracted of same detection region but having their contour slightly simplified to improve efficiency during the comparison process.

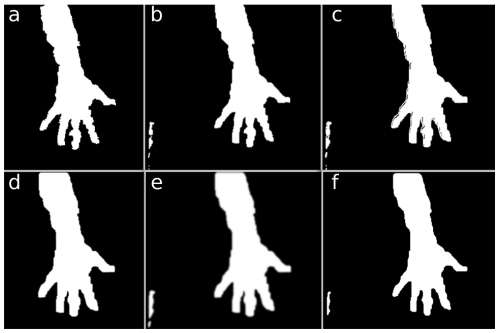


Fig. 3. Overview of the stages of preprocessing: (a) Image with noise; (b) Image of simplified contour; (c) Image after apply the filling algorithm; (d) Image after apply the smoothing algorithm; (e) Image after the Gaussian blur filter using a 7x7 window; (f) Final results after preprocessing.

The comparison calculation between the two images follows the negated exclusive disjunction operation where identical pixels generates true results and different pixels false ones. This operation was applied used the sum of the pixels of the two images producing a new image I_{dj} given by:

$$I_{dj}(x,y) = (I_{obj}(x,y) + I_{targ}(x,y)) \tag{6}$$

Values equal to zero indicate the true negative (tn) equal to 1 represent the false positives and negatives (fpn) and 2 is the number of true positives (tp). Accuracy (ACC) of the model is calculated by:

$$ACC = \frac{tp + tn}{tp + tn + fpn} \tag{7}$$

Due to inconsistency of the hand contour caused by noise is difficult to obtain exact match between the two images. Therefore an image is considered similar to another when ACC exceeds a threshold *P*. Noting the various comparisons *P* was defined as *0.9*.

C. Contourn analysis

The approaches convex hull and convex defects were used to determine the region of fingers and palm in the image. From contour analysis of hand the convexity can be found as the points located at more outer regions which was defined by seven points (figure 4b). By using each finger tip with the two wrist points the procedure for identifying the center of the palm and fingers is less complex.

For to complete this stage successfully is necessary the hand is always open and the fingers fully stretched for found all convexity points and consequently the fingers tips.

The convex hull and convex defect algorithm was enough to detect the location of finger tip and its base (figure 4c) but there are cases which the contour generate convex points not appropriate to fingertips due the instabilities on the edges

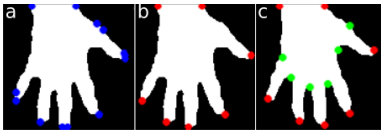


Fig. 4. Convexities of the hand. (a) Results after convex hull; (b) Simplified convex hull points by the method found in [11]; (c) Final results after finding the binary image of the hand with convex hull and convex defects.

therefore was necessary to apply the simplification algorithm used in section II-A. The final process is demonstrated in the figure 4.

D. Kinematics

The kinematic structure was found by considerate all centers of the joints of fingers in conjunction with the hand center (figure 5g). The joints of fingers was extracted from medial axis of each finger while the center of hand was identify using the highest pixel value after applying the EDT.

1) Medial axis of hand: To estimate the center of the phalanges was necessary to extract the medial axis of the hand. To improve the performance was necessary eliminate regions near edge reducing the amount of data to be processed by thinning morphological operation. Sharpen and blur median filters were applied on the picture aftert to apply EDT aim to maximize the difference between the center of hand and its contour and therefore eliminate unnecessary regions by applying a segmentation by threshold. The final result is achieved by applying the method in [12].

2) Finger medial axis extraction: The extraction of the branch for each finger was possible by applying an algorithm to select points of interest over all points resultants from the thinning algorithm and selecting the five first branches (left to right) and discarding the others which not correspond to the fingers (figure 6a).

To refine each branch to match the respective size of each finger was necessary identify the exact limits between dorsal side of the hand and the finger body. This was reached by calculate the midpoint among convex defects points and posteriorly using its location to find the smaller EDT relative to its respective branch. After defined all sizes of fingers the localization of phalanges was identified using statistical values proposed by [13].

E. Tracking

The proposed hand tracking was achieved by matching points between a group of four image of the hand in distinct poses and the hand image after the detection process both processed by EDT and median blur filter respectively. The algorithm ORB (a fusion of FAST keypoint detector and BRIEF descriptor) was used to match the points of interest among the images and the region of image containing the largest number of corresponding points was defined as initial location. This algorithm was chosen due the low computation

cost in relative to others algorithms like [14] and [15]. The final location was identified as the position of the pixel with most value within a box (*120x120*) applied at center of initial localization. This size box corresponds to hand size acquired during the detection process described in the II-B section. The principal component analysis was used to calculate the tilt angle of the hand relative to axis of sensor image plane. A part from this approaches was possible to track the hand and its kinematics.

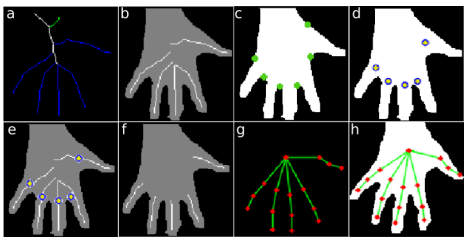


Fig. 5. (a) Extraction process of the medial axis of each finger; (b) Medial axis of the fingers; (c) Green dots indicate the convex defects points; (d) Yellow dots with blue edges indicate the midpoints of the defects of convexity; (e) Midpoints and the branchs to define the exactly location to eliminate the dorsal side of the branchs; (f) Approximate size of each finger; (g) The final kinematics structure; (h) The kinematics structure on the hand where the red dots indicate the phalanges and the hand center.

III. Preliminary Results

In this work two solutions were presented for preprocessing stage in order to reduce the noise presented on the edges in images captured by an RGB-D camera. These approaches were based on the relocation of points in the contour of hand (binary image) smoothing only the critical regions avoiding filter algorithm on the entire image. We also presented a method to estimate the approximate location of each phalanx using statistic values representing the approximate position of phalanx on the body of each finger. Finally it was possible to build a kinematic model of the hand that could be used to track the fingers. The tracking of fingers presented is invariant to rotation in relative to axis plan of the sensor and there are few flaws even with presence of other objects in the scene. Critical points were observed when the hands were situated in regions close to the border of the image.

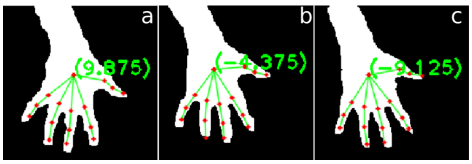


Fig. 6. Results from applying of the PCA algorithm on the binary image to detect the rotation in relative to axis plan of the sensor.

IV. Conclusion

In this paper a method was presented to detection of finger phalanges in RGB-D images. A kinematic model presented was restricted to make a rough estimate of the open hand position providing an initial framework for a more robust and accurate further processing to predict the full independent movement of each finger. The rotation of all the kinematic structure of a hand is shown in figure 6. To future work we intend to improve the hand tracking by using a descriptor of features with correspondence between similar points beyond consider the kinematic of various postures of the hand.

References

[1] F. Ricci, C. Perez, M. Fonseca, E. Guirro, and P. Santiago, "Protocolo experimental para análise cinemática da mão durante a utilização de órteses para membro superior," in *XXIV Congresso Brasileiro de Engenharia Biomédica*, June 2014, pp. 1993–1996.

[2] J. de Abreu, A. P. Fontana, and L. Menegaldo, "Reconstrução da cinemática da mão em pacientes com hanseníase," in *XXIV Congresso Brasileiro de Engenharia Biomédica*, Oct 2014, pp. 357–360.

[3] R. Y. Wang and J. Popović, "Real-time hand-tracking with a color glove," in *ACM SIGGRAPH 2009 Papers*, ser. SIGGRAPH '09, New York, NY, USA, 2009, pp. 63:1–63:8.

[4] A. P. Vicente and A. Faisal, "Calibration of kinematic body sensor networks: Kinect-based gauging of data gloves in the wild," in *Body Sensor Networks (BSN), 2013 IEEE International Conference on*, May 2013, pp. 1–6.

[5] M. Schroder, J. Maycock, H. Ritter, and M. Botsch, "Real-time hand tracking using synergistic inverse kinematics," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, May 2014, pp. 5447–5454.

[6] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Trans. Graph.*, vol. 33, no. 5, pp. 169:1–169:10, Sep. 2014.

[7] Y. Ouedraogo and Y. Aoki, "Finger posture estimation using 3d medial axes," in *Human System Interactions (HSI), 2014 7th International Conference on*, June 2014, pp. 71–75.

[8] A. Samadani, D. Kulic, and R. Gorbet, "Multi-constrained inverse kinematics for the human hand," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, Aug 2012, pp. 6780–6784.

[9] C. Metcalf, R. Robinson, A. Malpass, T. Bogle, T. Dell, C. Harris, and S. Demain, "Markerless motion capture and measurement of hand kinematics: Validation and application to home-based upper limb rehabilitation," *Biomedical Engineering, IEEE Transactions on*, vol. 60, no. 8, pp. 2184–2192, Aug 2013.

[10] S. Suzuki and K. be, "Topological structural analysis of digitized binary images by border following," *Computer Vision, Graphics, and Image Processing*, vol. 30, no. 1, pp. 32 – 46, 1985.

[11] U. Ramer, "An iterative procedure for the polygonal approximation of plane curves," *Computer Graphics and Image Processing*, vol. 1, no. 3, pp. 244 – 256, 1972.

[12] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Commun. ACM*, vol. 27, no. 3, pp. 236–239, Mar. 1984.

[13] A. Buryanov and V. Kotiuk, "Proportions of Hand Segments," *International Journal of Morphology*, vol. 28, pp. 755 – 758, 09 2010.

[14] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision 2*, 1999, p. 11501157.

[15] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proceedings of the ninth European Conference on Computer Vision*, May 2006.

TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA DIGITAL DA UFBA

1 Identificação do tipo de documento

Tese [ ] Dissertação [x] Monografia [ ] Trabalho de Conclusão de Curso [ ]

2 Identificação do autor e do documento

Nome completo: Thalisson Nobre Santos

CPF: 031.301.305-51

Telefone: (71) 99253-1347 e-mail: thalisson.ns@gmail.com

Programa/Curso de Pós-Graduação/Graduação/Especialização: Mestrado em mecatrônica

Título do documento: Detecção e rastreamento da mão utilizando dados de profundidade

Data da defesa: 16/12/2015

3 Autorização para publicação na Biblioteca Digital da UFBA

Autorizo com base no disposto na Lei Federal nº 9.610, de 19 de fevereiro de 1998 e na Lei n o 10.973, de 2 de dezembro de 2004, a Universidade Federal da Bahia (UFBA) disponibilizar gratuitamente sem ressarcimento dos direitos autorais, o documento supracitado, de minha autoria, na Biblioteca Digital da UFBA para fins de leitura e/ou impressão pela Internet a título de divulgação da produção científica gerada pela Universidade.

Texto completo [x] Texto parcial [ ]

Em caso de autorização parcial, especifique a (s) parte(s) do texto que deverão ser disponibilizadas:

Salvador, 16/02/2016
Local Data

Thalisson Nobre Santos
Assinatura do (a) autor (a) ou seu representante legal

4 Restrições de acesso ao documento

Documento confidencial? [x] Não

[ ] Sim Justifique: Informe a data a partir da qual poderá ser disponibilizado na Biblioteca Digital da UFBA:

__/__/____ [x] Sem previsão

Assinatura do Orientador: (Opcional)

O documento está sujeito ao registro de patente? Não [x]
Sim [ ]

O documento pode vir a ser publicado como livro? Sim [x]
Não [ ]

Preencher em três vias. A primeira via deste formulário deve ser encaminhada ao Sistema de Bibliotecas da UFBA/Biblioteca Central; a segunda deve ser enviada para a Biblioteca de sua Unidade, juntamente com o arquivo contendo o documento; a terceira via deve permanecer no Programa de Pós-Graduação para o registro do certificado de conclusão do Curso.

Universidade Federal da Bahia
Sistema de Biblioteca da UFBA
Grupo Técnico da Biblioteca Digital da UFBA

CADASTRO DE INFORMAÇÕES PARA PUBLICAÇÃO DIGITAL
NA BIBLIOTECA DIGITAL DA UFBA

1. Identificação do tipo de material	
Tese ( ) Dissertação (x) Monografia ( ) Trabalho de Conclusão de Curso ( )	
2 . Colegiado do Curso de Pós-Graduação:	
Título: Detecção e rastreamento da mão utilizando dados de profundidade	
Autor(a): Thalisson Nobre Santos	
CPF: 031.301.305-51	*E-mail:* thalisson.ns@gmail.com
Orientador(a): :	
Nome: Prof. Dr. Luciano Rebouças de Oliveira	
CPF: 673.235.295-49	*E-mail:* lreboucas@ufba.br
Co-Orientadores	
Nome:	
CPF:	*E-mail:*

Membros da Banca	
Nome: Prof. Dr. Luciano Rebouças de Oliveira (Orientador)	
CPF: 673.235.295-49	*E-mail:* lreboucas@ufba.br
Nome: Prof. Dr. Maurício Pamplona Segundo	
CPF: 058.732.739-10	*E-mail:* mauricio@dcc.ufba.br
Nome: Profa. Dra. Michele Fúlvia Angelo	
CPF: 254.540.938-85	*E-mail:* mfangelo@ecomp.uefs.br
Nome:	
CPF:	*E-mail:*

Data de Homologação Pós-Graduação:	
Financiadores:	
Data:	
Assinatura: Thalisson Nobre Santos	

Universidade Federal da Bahia
Sistema de Biblioteca da UFBA
Grupo Técnico da Biblioteca Digital da UFBA

Salvador, 16/02/2016

DECLARAÇÃO

Declaro para os devidos fins que o texto final apresentado para a conclusão do meu curso de Mestrado em Mecatrônica da Universidade Federal da Bahia é de minha autoria. Declaro também que quaisquer informações utilizadas neste texto, mas que sejam provenientes de outros trabalhos tem fonte claramente expressa e, quando for o caso, foram devidamente autorizadas pelo(s) respectivo(s) autor(es).

Thalisson Nobre Santos

Nome: Thalisson Nobre Santos
CPF: 031.301.305-51