DISSERTAÇÃO DE MESTRADO

UNIVERSIDADE FEDERAL DE SÃO CARLOS

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM

CIÊNCIA DA COMPUTAÇÃO

"Reconhecimento de gestos da Língua Brasileira de Sinais através de Máquinas de Vetores de Suporte e Campos Aleatórios Condicionais Ocultos"

ALUNO: César Roberto de Souza
ORIENTADOR: Prof. Dr. Ednaldo Brigante Pizzolato

São Carlos
Abril/2013

CAIXA POSTAL 676
FONE/FAX: (16) 3351-8233
13565-905 - SÃO CARLOS - SP
BRASIL

César Roberto de Souza

Reconhecimento de gestos da Língua Brasileira de Sinais através de Máquinas de Vetores de Suporte e Campos Aleatórios Condicionais Ocultos

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Computação da Universidade Federal de São Carlos como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Área de Concentração:
Processamento de Imagens e Sinais
Orientador:
Prof. Dr. Ednaldo Brigante Pizzolato

São Carlos

2013

Ficha catalográfica elaborada pelo DePT da Biblioteca Comunitária da UFSCar

S729rg

Souza, César Roberto de.
Reconhecimento de gestos da língua brasileira de sinais através de máquinas de vetores de suporte e campos aleatórios condicionais ocultos / César Roberto de Souza. -- São Carlos : UFSCar, 2013.
218 p.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2013.

1. Processamento de imagens. 2. Reconhecimento de padrões. 3. Visão computacional. 4. Linguagem por sinais. I. Título.

CDD: 006.42 (20ª)

Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

Programa de Pós-Graduação em Ciência da Computação

"Reconhecimento de Gestos da Língua Brasileira de Sinais Através de Máquinas de Vetores de Suporte e Campos Aleatórios Condicionais Ocultos"

César Roberto de Souza

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação

Membros da Banca:

Prof. Dr. Ednaldo Brigante Pizzolato
(Orientador - DC/UFSCar)

Prof. Dr. Nelson Delfino d'Ávila Mascarenhas
(DC/UFSCar)

Prof. Dr. Roberto Hirata Junior
(IME/USP)

São Carlos
Maio/2013

Agradecimentos

Agradeço primeiro a minha família, por todo o suporte, esforço e dedicação que me proporcionaram ao longo destes anos, especialmente a meus pais Caio e Clotilde, que tanto me apoiaram, mesmo diante de tantas dificuldades. Agradeço demais a meus irmãos Cristiano, Cassius, e minha irmã Cláudia, que tantas vezes me ofereceram sua hospitalidade e compreensão por mais que morássemos tão longe, e a meu irmão Caio, pelo apoio enquanto terminava este curso de Mestrado.

Agradeço ao professor, orientador e amigo Dr. Ednaldo Brigante Pizzolato pela base, amparo e, principalmente, pelas inúmeras críticas feitas durante o período de trabalho realizado. É sempre difícil reconhecer nossos próprios erros sem uma segunda visão experiente sobre o assunto.

Agradeço também a meus amigos, Guilherme Cartacho Pedroso pela ajuda na compreensão dos modelos ocultos de Markov, ainda em 2008, e por todo o trabalho coletando as mais de 90 mil amostras que compõem seu banco de dados original de imagens de intensidade; a Mauro Dos Santos Anjo, pelas inúmeras discussões, dicas e toques sobre processamento de imagens e reconhecimento de padrões em geral, bem como toda a ajuda prestada em compreender seus trabalhos anteriores e pela base fornecida para que esta pesquisa caminhasse seu próprio caminho; e a todos meus colegas de mestrado e de república, por toda diversão, discussões e ideias partilhadas em durante nosso trajeto.

Um agradecimento especial ao professor Dr. Alexandre Levada pelas inúmeras conversas e discussões acerca de processamento de imagens e teoria dos grafos; ao Dr. Estevam Rafael Hruschka Júnior, pelas compreensivas aulas em aprendizado de máquina; e ao Dr. Nelson Delfino d'Ávila Mascarenhas pelas aulas de reconhecimento de padrões e processamento de imagens que sempre serão lembradas.

Ofereço meus agradecimentos ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela bolsa concedida durante estes dois anos, sem a qual seria impossível conduzir esta pesquisa.

Agradeço a Deus por ter colocado todas estas pessoas em minha vida.

"É impossível àqueles que não entendem a Língua dos Sinais compreender suas possibilidades ao surdo, sua influência poderosa na felicidade moral e social destes desprovidos da audição, e seu maravilhoso poder de carregar pensamentos a intelectos que de outra maneira estariam em escuridão perpétua. Nem podem estes apreciar o domínio que tem sobre os surdos. Enquanto houver dois surdos na face da Terra e estes dois se encontrem, haverá o uso dos sinais."

— Joseph Schuyler Long (The sign language: a manual of signs, 1910)

Resumo

Este trabalho investiga o uso de Máquinas de Vetores de Suporte e Campos Condicionais Aleatórios Ocultos no problema de reconhecimento de sinais pertencentes à Língua Brasileira de Sinais (Libras). A partir da aplicação dos preceitos básicos da Teoria de Aprendizado Estatístico de Vapnik, o problema do reconhecimento de gestos na Libras foi fundamentado como um caso de aprendizado supervisionado definido sobre imagens e sequências de imagens, evitando-se ao máximo o mal condicionamento tipicamente encontrado na estimação de densidades através da utilização de modelos discriminativos. Partindo-se de estudos linguísticos sobre a formação do sinal da Libras, foi criada uma arquitetura de classificação baseada em duas camadas operando sobre características extraídas de imagens capturadas através de sensores (câmeras) de profundidade. Foram adotadas abordagens quantitativas para a comparação e análise de resultados através do uso de tabelas de contingência e testes estatísticos de hipótese, que se verificam estatisticamente significantes favorecendo a escolha de modelos aqui apresentada. Dentre os resultados encontrados, foi encontrado nas Máquinas de Vetores de Suporte organizadas em Grafos Direcionais Acíclicos o balanço necessário entre eficiência e acurácia para a classificação de estruturas sublexicais da Libras, enquanto os Campos Aleatórios Condicionais Ocultos forneceram um aumento nas taxas de reconhecimento de palavras e sinais da Libras sem que para isto tivessem de comprometer a capacidade de generalização do sistema aqui apresentado.

Palavras-chave: língua de sinais, segmentação de imagens, reconhecimento de padrões, reconhecimento de sequências, máquinas de vetores de suporte, campos condicionais aleatórios.

Abstract

This work investigates the use of Support Vector Machines and Hidden Conditional Random Fields in the recognition of signs from the Brazilian Sign Language (*Língua Brasileira de Sinais*, Libras). Employing basic concepts from Vapnik's Statistical Learning Theory, the gesture recognition problem is cast as a supervised learning problem defined over images and image streams, avoiding the inherent ill-conditioning present in many density estimation problems through the use of discriminative classification models. From linguistic studies on the structural formation of the Libras sign, a two-layer recognition architecture has been created to operate over features extracted from depth images captured through a depth sensor. This work utilizes quantitative approaches for performance assessment, performing comparisons through contingency tables and statistical hypothesis tests; revealing statistically significant results favoring the aforementioned choice of classification models. Results have shown how the multiclass SVMs organized in Directed Acyclic Graphs provided a needed balance between efficiency and accuracy in the classification of the sub-lexical structures of the Libras, whereas Hidden Conditional Random Fields boosted the system's recognition rates without for this sacrificing its generalization for unobserved instances.

Keywords: Sign languages, image segmentation, human-computer interaction, pattern recognition, support vector machines, conditional random fields.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

ANN	Rede Neural Artificial (*Artificial Neural Network*)
ASL	Língua de Sinais Americana (*American Sign Language*)
BSL	Língua de Sinais Britânica (*British Sign Language*)
CM	Configuração de Mãos
CV	Validação Cruzada (*Cross-Validation*)
CRF	Campos Aleatórios Condicionais (*Conditional Random Fields*)
DAG	Grafo Direcionado Acíclico (*Directed Acyclic Graph*)
DT	Árvore de Decisão (*Decision Tree*)
GS	Busca em grade (*Grid-Search*)
i.i.d	Independentes e Identicamente Distribuídas
HCRF	Campos Aleatórios Condicionais Ocultos (*Hidden Conditional Random Fields*)
HMM	Modelos Ocultos de Markov (*Hidden Markov Model*)
ICA	Análise de Componente Independente (*Independent Component Analysis*)
LDCRF	Campos Aleatórios Condicionais Latente-Dinâmicos (*Latent-Dynamical Conditional Random Fields*)
LIBRAS	Língua Brasileira de Sinais
LMA	Algoritmo de Levenberg Marquardt (*Levenberg-Marquardt Algorithm*)
MAP	Máximo a Posteriori (*Maximum a Posteriori*)
ML	Máxima Verossimilhança (*Maximum Likelihood*)
MRF	Campo Aleatório Markoviano (*Markov Random Field*)

Rprop	Retropropagação Resiliente (*Resilient Backpropagation*)
SMO	Otimização Sequencial Mínima (*Sequential Minimal Optimization*)
SRM	Minimização do Risco Estrutural (*Structural Risk Minimization*)
SVM	Máquinas de Vetores de Suporte (*Support Vector Machine*)
SV	Vetores de Suporte (*Support Vectors*)

Sumário

Parte I

Fundamentação

Capítulo 1

Introdução

"E era toda a terra de uma mesma língua e de uma mesma fala." — Genesis 11:1.

As formas de comunicação vão muito além do que pode ser simplesmente lido, escrito, falado ou ouvido. Mesmo muitas vezes passando despercebidos ao consciente, os gestos desempenham papel integrante, senão fundamental, na completa expressão de nossas ideias ao engajarmos um diálogo, conversação, ou qualquer tarefa de comunicação em que exista o contato visual entre o locutor e o ouvinte. No caso do canal de comunicação auditivo-fonológico encontrar-se indisponível ou danificado, os gestos passam a outro patamar: tornam-se a via principal de comunicação e expressão da comunidade surda através das Línguas de Sinais.

Estudando apenas gestos articulados junto da fala, pode-se dizer que apenas cerca de 30 a 35% do significado social de uma conversa ou interação seja realmente transmitido através de palavras (BIRDWHISTELL, 1970, p. 158), sendo todo o restante transmitido através da comunicação não-verbal. Mas é quando analisamos os gestos no contexto das Línguas de Sinais é que se pode compreender a magnitude da importância que desempenham na vida social dos que não podem usufruir da fala ou da audição. O uso de gestos como elementos visuais nas Línguas de Sinais, como é exemplo da Língua Brasileira de Sinais (Libras), possibilita que o deficiente auditivo tenha uma porta aberta para sua inclusão na sociedade.

Ao contrário do que uma vez se acreditara, as Línguas de Sinais não constituem mímica; tampouco são apenas *linguagens*, no sentido de serem diferentes das Línguas Orais. As Línguas de Sinais, como a Libras, são exemplos completos de línguas naturais. Não nos referimos às línguas portuguesa, inglesa ou francesa apenas como *linguagens*. Não seria correto, portanto, nos referir às Línguas de Sinais como apenas *linguagens* de sinais, já que estas não estão qualificadas nem acima nem abaixo das outras línguas naturais.

Uma vez que as Línguas de Sinais constituem a principal via de acesso ao universo da cultura surda (STROBEL, 2008), a busca na elaboração de um sistema

computacional capaz de ajudar na interpretação e, potencialmente, *traduzir* a Língua de Sinais é de grande importância social e tecnológica. Tendo em vista todos os potenciais benefícios e contribuições oriundas da elaboração de um destes sistemas, é evidente o quanto o desenvolvimento de um sistema desta categoria possa melhorar tanto a vida social do deficiente auditivo, quanto às descobertas encontradas durante seu desenvolvimento possam contribuir tanto as áreas de computação, teoria linguística e das neurociências.

1.1 Contexto

Este trabalho está inserido num contexto multidisciplinar abrangendo subáreas da Ciência da Computação e da Linguística. Mesclando a teoria Linguística do estudo das línguas gestuais e das demais línguas naturais com o processamento de imagens e o reconhecimento de padrões, este trabalho tenta reunir várias técnicas e resultados aplicáveis para possibilitar o desenvolvimento de um sistema reconhecedor da Língua de Sinais.

As línguas baseadas em gestos são tão ricas e diversas quanto qualquer outra língua natural. Ao contrário do popularmente acreditado, as Línguas de Sinais não são universais – na data de elaboração deste trabalho, o catálogo de linguagens vivas *Ethnologue* (LEWIS, 2009) registrava cerca de 130 linguagens de sinais existentes no mundo. Nota-se, no entanto, que este valor deve-se tratar de uma subestimativa, já que o número de linguagens registradas no catálogo vem crescendo ao longo dos anos. Novas linguagens podem surgir espontaneamente, como quando tomamos, por exemplo, o ocorrido registrado na Nicarágua durante a recente geração da Língua Gestual Nicaraguense (SENGHAS e ÖZYÜREK, 2004). Exemplificando tamanha diversidade dos sinais gestuais existentes, pode-se caracterizar o sorriso como um dos únicos gestos existentes capaz de ser entendido da mesma maneira em todos os locais do mundo (EKMAN, 1993).

Na Linguística, vasto estudo tem sido conduzido visando entender, catalogar, modelar e caracterizar a linguagem dos gestos, linguagem esta não apenas restrita às línguas gestuais (FERREIRA-BRITO, 2010; LUCAS, 2001; O'BRIEN, 2005; STOKOE, 2001; LIRA e SOUZA, 2008), mas também aos gestos articulados com a fala através do estudo da Cinésica (BIRDWHISTELL, 1970; SUTTON-SPENCE, 1999; VALLI e LUCAS, 2000; KENDON, 2004). Na Ciência da Computação, principalmente na área interdisciplinar de Interação Humano-Computador, o

uso de gestos se torna particularmente atraente por incorporar expressões naturais na interface de interação com sistemas computacionais. Exemplos deste uso datam desde a década de 80 (BOLT, 1980), e veem se popularizando ao longo dos anos, aparecendo em inúmeros trabalhos recentes, como os diversos trabalhos conduzidos por Feuerstack e colegas (FEUERSTACK, COLNAGO e SOUZA, 2011; FEUERSTACK, ANJO e PIZZOLATO, 2011; ANJO, PIZZOLATO e FEUERSTACK, 2012).

No entanto, mesmo o uso dos gestos não sendo recente na Ciência da Computação — as primeiras interfaces baseadas em gestos datando de tempos em que o poder computacional dos dispositivos de hoje não eram sequer encontrados em supercomputadores (BOLT, 1980) —, a trilha em busca de modelos computacionais adequados, estratégias para tornar o problema tratável e diferentes abordagens para simplificar ou remover restrições destes sistemas é válida até hoje, e dificilmente pode-se considerar o problema de reconhecimento de gestos um problema resolvido na computação.

1.2 Motivação e Objetivos

A motivação deste trabalho é composta principalmente por duas frentes: a motivação de cunho social, na busca de um sistema capaz de aumentar a inclusão social do surdo, principalmente no Brasil; e a motivação de cunho científico, na investigação das diferentes maneiras de interação, dos modelos computacionais empregados e estudados durante esta jornada, juntamente com seus respectivos desafios. Mais do que isso, pode-se notar que o entendimento da língua dos gestos tem se mostrado interessante no auxílio da compreensão das demais línguas naturais (LUCAS, 2001). A busca por modelos computacionais que possam interpretar a língua gestual torna-se uma tarefa bastante interessante por unificar diversas áreas do conhecimento, desde a inteligência artificial, a estatística, e o processamento de imagens até a linguística e as neurociências. A obtenção destes modelos e resultados, por sua vez, podem trazer colaborações interessantes tanto de cunho social, no caso da inclusão de deficientes auditivos, quanto à aprimoração das teorias e de modelos linguísticos e cognitivos disponíveis.

No Brasil, a Libras é a língua oficial disponível aos deficientes auditivos, reconhecida desde o ano de 2002 após a instauração da lei n° 10.436 de 24 de abril de 2002 e o decreto n° 5626 de 22 de dezembro de 2005 que a define e a regulamenta.

Sendo a Libras a língua gestual oficial do país, objetivamos com este trabalho colaborar com a pesquisa em meios de acessibilidade que sejam capazes de aumentar a inclusão social dos mais de 9.717.318 (IBGE, 2010) deficientes auditivos brasileiros[1]. Neste trabalho, abordaremos *alguns* dos gestos presentes na Libras e investigaremos uma possível abordagem para o problema. Note-se, no entanto, que não seremos pretenciosos o bastante a ponto de propor uma solução única e final para o problema, mas almejaremos trata-lo com o mínimo de restrições possíveis. Desta maneira, evitaremos o uso de luvas especiais, sejam elas eletrônico-sensoriais ou simplesmente coloridas, e evitaremos o uso de hipóteses severamente restritivas sobre as vestimentas do usuário, como uso de jaquetas ou mangas compridas ou de cores específicas. Buscaremos, assim, abordar uma característica específica da Libras sem que as hipóteses assumidas potencialmente prejudiquem sua generalização para características mais completas ou elaboradas.

1.3 Contribuições e limitações

A principal contribuição deste trabalho será a investigação de maneiras de representar e detectar sinais da Libras em uma aplicação de reconhecimento de imagens cujo ambiente de fundo possa ser arbitrariamente complexo. Mais especificamente, estaremos contribuindo com:

- Um algoritmo de segmentação capaz de extrair características conjuntas da face e das mãos durante a articulação do sinal na Libras, construindo sobre o estudo de Anjo (2012);
- A avaliação da aplicabilidade do uso de Máquinas de Vetores de Suporte ao problema de reconhecimento de gestos estáticos e um estudo acerca de sua extensibilidade a um crescente número de características à medida que avançamos no reconhecimento de sinais da Libras;
- A avaliação da aplicabilidade do uso de Campos Aleatórios Condicionais Ocultos ao problema de reconhecimento de gestos dinâmicos;

[1] Destes, 344.206 declararam-se completamente incapazes de ouvir; 1.798.967 relatam grande dificuldade em audição e 7.574.145 alguma dificuldade. Em comparação com o censo anterior (IBGE, 2003), tínhamos 5.735.099 deficientes auditivos – apesar de esta diferença poder ser parcialmente explicada devido a diferenças na metodologia de pesquisa entre os diferentes censos.

Como evidenciam Nicoloso e Silva (2009) mesmo intérpretes humanos enfrentam muitos desafios na interpretação simultânea em situações em que interrupções ao usuário de língua de sinais para recuperar uma informação perdida não são viáveis ou possíveis, o que é um indicativo de que a elaboração completa de um destes sistemas pode estar fora do alcance de uma dissertação de mestrado. Assim, sem maior demora, definiremos as limitações deste trabalho. Neste trabalho, assumiremos as seguintes hipóteses simplificadoras:

- Uma única pessoa estará diante de uma câmera em um dado momento;
- O espaço de articulação ao redor da pessoa deve conter apenas a pessoa;
- Condições de iluminação no espaço de articulação serão adequadas;
- O sensor de profundidade estará em uma posição fixa, diante do usuário.

E especificamente, **não** estaremos supondo:

- Uma vestimenta específica para auxiliar as etapas de segmentação,
- Luvas ou quaisquer outros dispositivos que devam ser usados pelo usuário.

Neste trabalho, abordaremos apenas *alguns* dos gestos da Libras e investigaremos uma possível abordagem para o problema. De maneira nenhuma buscaremos *tratar todos os possíveis gestos e variações da língua*, mas gravitaremos acerca de métodos que tenham pelo menos a possibilidade de se estender e acomodar novos destes gestos e variações.

1.4 Organização do trabalho

Neste trabalho, exploraremos formas de detecção de gestos específicos da Libras envolvendo mais do que o simples alfabeto manual. Seguir-se-á então um relato das abordagens e técnicas exploradas na obtenção de tal sistema. Este documento está divido em nove capítulos, separados em duas partes.

A primeira parte é composta da fundamentação teórica, como levantamento bibliográfico e apresentação dos métodos a serem utilizados. No Capítulo 2, apresentaremos uma revisão bibliográfica, contemplando desde a detecção e segmentação do corpo humano em imagens coloridas e de profundidade até o desenvolvimento de interfaces gestuais, reconhecimento de gestos e expressões faciais. No Capítulo 3, enunciaremos as principais características da Língua Brasileira de Sinais (Libras),

incluindo investigações conduzidas na Ciência Linguística direcionadas à especificação e caracterização de sua gramática. No Capítulo 4, discutiremos as principais técnicas de processamento de imagens disponíveis para abordar este problema. No Capítulo 5, apresentaremos como será possível lidar com este problema de maneira automática a partir de técnicas de reconhecimento de padrões.

A segunda parte deste trabalho contém o desenvolvimento do trabalho em si, apresentando a abordagem utilizada e os experimentos realizados. A partir do Capítulo 6 detalhamos a abordagem utilizada para se atacar problemas específicos da Libras, seguida da metodologia de pesquisa utilizada no Capítulo 7, e os experimentos conduzidos e seus resultados no Capítulo 8. Por fim, apresentamos as principais conclusões deste trabalho no Capítulo 9, fornecendo indicações para trabalhos futuros, fechando o escopo desta dissertação.

Este documento contem ainda três apêndices: um contendo o termo de consentimento livre e esclarecido apresentado aos participantes dos experimentos realizados; outro contendo detalhes dos experimentos realizados no Capítulo 8; e um terceiro apresentando o ferramental utilizado no desenvolvimento desta pesquisa.

Capítulo 2

Revisão Bibliográfica

"O homem pode tudo quanto sabe." — Francis Bacon

Na ciência, quase tão importante quanto realizar nossas próprias descobertas, é revisar e retomar as descobertas realizadas pelos que vieram antes de nós. Apenas assim é possível conhecer as paisagens já deslumbradas pelos que se aventuraram em explorar este mesmo campo, e assim delimitar as fronteiras e limitações destes trabalhos anteriores, a fim de se contribuir com um próximo passo no conhecimento científico. Desta maneira, significante parte deste trabalho se dedicará agora ao levantamento bibliográfico e delineamento geral das fronteiras exploradas na área do reconhecimento de gestos até então.

É curioso observar que até presente data poucos trabalhos se dedicaram a estudar a Libras do ponto de vista da elaboração de um sistema de reconhecimento automático de gestos. É mais curioso ainda observar que este problema se estende também a própria pesquisa em Linguística: como enunciam Nicoloso e Silva (2009), as investigações em torno da Libras vem ocorrendo apenas timidamente, dificultando a pesquisa na área devido à escassez de publicações. A elaboração de uma gramática, cuja necessidade foi levantada por Ferreira Brito (2010), e sua importância na criação de sistemas de reconhecimento automático de Línguas de Sinais, conforme evidenciada por Starner (1995), também parece ter recebido pouca atenção dentre pesquisas anteriores.

Na literatura, gestos têm sido tradicionalmente separados em duas categorias: gestos estáticos e gestos dinâmicos (MITRA e ACHARYA, 2007). Gestos são considerados *estáticos* quando não envolvem movimento, mas apenas uma pose ou configuração. Gestos *dinâmicos*, em contrapartida, envolvem movimento e trajetória específicos. É natural observar que estas duas categorias não são mutuamente ex-

cludentes, e a maioria dos gestos, de fato, envolve ambos componentes estáticos e dinâmicos.

Como constata a pesquisa de Mitra e Acharya (2007), pode-se dizer que gestos em geral são ambíguos e incompletamente especificados. Muitos gestos diferentes podem ser utilizados para designar uma mesma coisa, como também é possível que gestos iguais tenham significados distintos em diferentes culturas. Entrando momentaneamente no contexto das línguas gestuais, pode-se dizer que, assim como na fala e na escrita, a execução de um gesto varia tanto de pessoa para pessoa como entre diferentes repetições realizadas por uma mesma pessoa.

Podemos ver que o problema de reconhecimento de gestos é um problema interessante dada sua não trivialidade e aos muitos aspectos dos diferentes subproblemas encontrados por quem se aventura na área. Sistemas de reconhecimento de gestos podem ser concebidos de muitas maneiras, e cada abordagem é caraterizada pelas restrições que impõem ao usuário. Abordagens baseadas no uso de sensores, por exemplo, como luvas eletrônicas ou outros dispositivos de rastreio (BRAFFORT, 1997; VOGLER e METAXAS, 2001; CHEN, FANG, *et al.*, 2004), limitam o usuário ao uso destes dispositivos específicos. Da mesma maneira, métodos baseados em visão computacional podem fazer uso da hipótese de que o usuário esteja utilizando luvas coloridas ou outros marcadores específicos. Os trabalhos também podem ser distinguidos quanto ao foco em sinais isolados (GROBEL e ASSAN, 1997; BRETZNER, LAPTEV e LINDEBERG, 2002; DIAS, MADEO, *et al.*, 2009; PIZZOLATO, ANJO e PEDROSO, 2010) ou sinais embutidos em sequências contínuas de padrões (YANG, SCLAROFF e LEE, 2009; STARNER, 1995), que é o caso mais geral nas Línguas de Sinais.

Neste trabalho, daremos ênfase a abordagens baseadas em visão computacional. Porém, mesmo estas técnicas podem variar significantemente entre si. Segundo Mitra e Acharya (2007), estas diferenças podem ocorrer quanto ao número de câmeras utilizadas, a velocidade e latência destas câmeras, as restrições do ambiente (como iluminação), restrições do usuário (como o uso de marcadores), a natureza das características utilizadas (como silhuetas, bordas, histogramas), se a informação utilizada é 2-D ou 3-D e, finalmente, se a abordagem utilizada incorpora a representação do tempo ou não. Dentro das técnicas citadas, daremos particular ênfase a abordagens que não imponham restrições ao usuário e que possuam mínimas restrições de ambiente. Como será necessário tratar gestos dinâmicos, adotaremos técnicas que envolvam representação do tempo, em particular técnicas baseadas em modelos ocultos de Markov (STARNER, 1995; STARNER, WEAVER e PENTLAND,

1998; ELMEZAIN, AL-HAMADI e MICHAELIS, 2009) e campos aleatórios condicionais (YANG, SCLAROFF e LEE, 2009; ELMEZAIN, 2010; YANG e SARKAR, 2006).

2.1 Interfaces Gestuais e o Reconhecimento de Gestos

A motivação para a criação de interfaces baseadas em gestos é bastante antiga. Um dos mais famosos experimentos na interpretação de gestos para interação humana-computador é dada pelo trabalho pioneiro de Richard Bolt (1980). Em sua demonstração, usuários manejavam formas simples em um grande painel visual utilizando apenas gestos e a fala, um fato notável levando-se em consideração a tecnologia disponível nesta época. No entanto, foram necessárias mais de três décadas de pesquisa até que interfaces baseadas em gestos ganhassem tração em sua popularidade. Durante esta época, inúmeros trabalhos abordaram o problema de diferentes maneiras, com diferentes abordagens e respectivas limitações. Gradualmente, espera-se que as limitações sejam removidas e a construção de uma via confiável de comunicação e de interpretação auxiliada por modelo computacional seja finalmente concebida.

O trabalho pioneiro em (BOLT, 1980) pré-datava a época em que interfaces gráficas (GUIs) eram comuns em dispositivos computacionais. A *Media Room* criada pelo *Architecture Machine Group* do Instituto de Tecnologia de Massachusetts (MIT), do tamanho de um escritório pessoal, contava com uma tela de retroprojeção ocupando toda a parede frontal, disposta a cerca de 2 metros do usuário. Suas paredes abrigavam alto-falantes em ambos os lados da tela de projeção, bem como em ambos os lados e atrás da cadeira do usuário operador. Esta sala servia como base de estudos para a pesquisa na construção de um Sistema de Gerenciamento de Dados Espacial (*Spatial Data-Management System*, SDMS). Os usuários podiam interagir com o ambiente virtual através de gestos, apontação ou referências espaciais simples como "*cima*", "*baixo*", e "*à esquerda/direita de*".

Sua sala de interação utilizava um componente de reconhecimento de fala conectada criado pela *NEC of America, Inc.* O reconhecimento de fala conectada era um problema clássico na época, e este sistema, por reconhecer a fala conectada, não necessitava de pausas entre as palavras e conseguia reconhecer sentenças de até cinco palavras faladas. O vocabulário deste sistema podia chegar a um máximo de 120 palavras, ou até cerca de 1000 palavras em modo de reconhecimento discreto.

Os gestos e apontamentos eram detectados com o auxílio de sensores magnéticos. O usuário deveria carregar um sensor, conectado por um fio, que poderia ser preso como anel ou bracelete. Como mencionado anteriormente, seu trabalho precedia a noção de área de trabalho comumente encontrada nos computadores pessoais de hoje, e, como tal, seu trabalho apresentava na época um grande leque de novas possibilidades.

Em paralelo à pesquisa em interfaces gestuais e multimodais, a pesquisa em reconhecimento de fala floresceu[2] na década de 80. O emprego das técnicas baseadas em casamento de modelos foi gradualmente substituído por técnicas baseadas em modelos estatísticos, particularmente os modelos ocultos de Markov (*Hidden Markov Models*, HMM), que viriam a se tornar o método preferido para reconhecimento de fala (JUANG e RABINER, 2005). No entanto, uma das primeiras, se não a primeira publicação a investigar a aplicabilidade de HMMs no problema de reconhecimento de gestos humanos ocorreu somente em 1992 com o trabalho de Yamato, Ohya e Ishii (1992). Seus experimentos se baseavam na classificação de seis tipos distintos de movimentos do jogo de tênis, apresentando cerca de 90% de acerto na rotulação de seu conjunto de dados. O método empregava a quantização vetorial do espaço de características para suprir símbolos discretos aos modelos de Markov, em contraste a utilização de modelos contínuos preferidos no campo de reconhecimento de fala.

Gradativamente, modelos ocultos de *Markov* foram ganhando maior popularidade na área de reconhecimento de gestos. Em 1995, um dos primeiros trabalhos a utilizar HMMs no reconhecimento de gestos em Línguas de Sinais foi o trabalho conduzido por Starner (1995), que abordaremos com mais detalhes na seção 2.2 (dedicada exclusivamente ao tema de reconhecimento de línguas de sinais). Em resumo, Starner utilizou HMMs com densidades de probabilidade contínuas modeladas através de misturas de Gaussianas multidimensionais. No entanto, uma dificuldade de seu trabalho devia-se ao fato de seu sistema não ser capaz de identificar quando uma sequência de movimentos realizada não era um gesto que deveria ser reconhecido pelo sistema. Em outras palavras, seu sistema não incorporava mecanismos para rejeição.

[2] Deve-se notar que, apesar dos avanços e a maturidade do campo de pesquisa, a área de reconhecimento e processamento de fala ainda hoje se encontra bastante defasada (DAHL, YU, *et al.*, 2012).

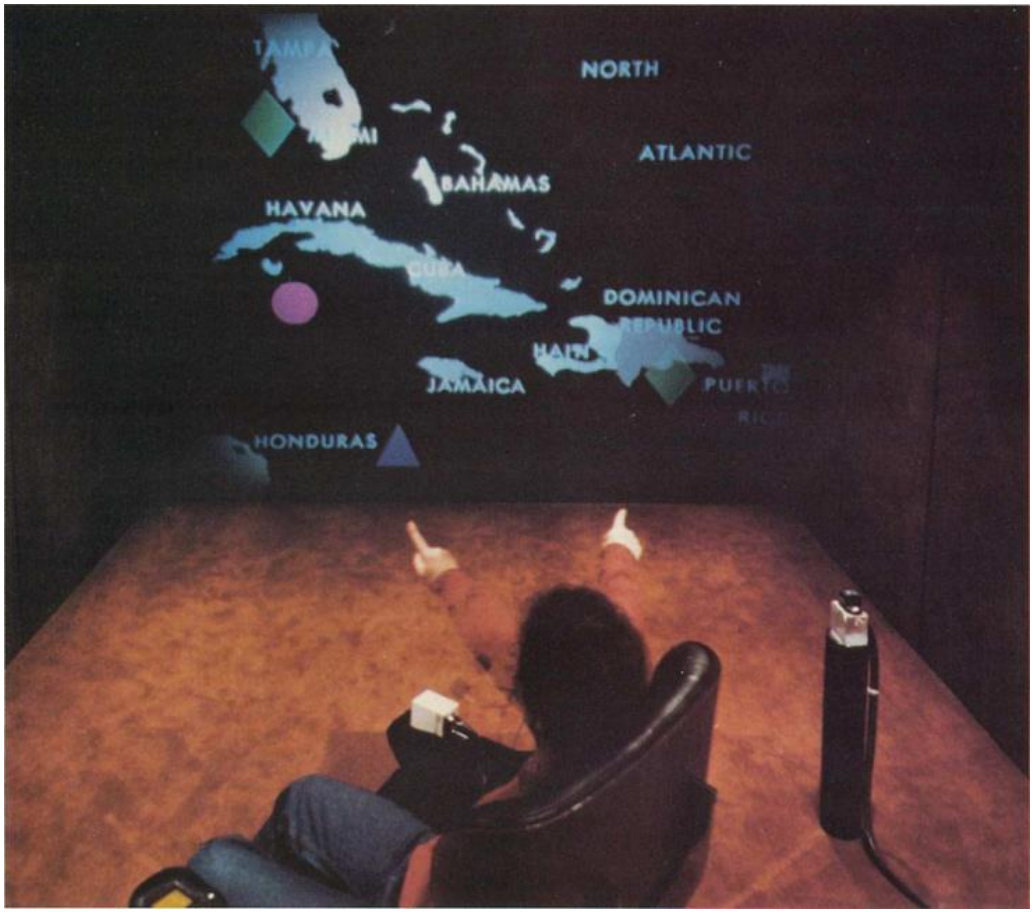

Figura 1. Demonstração do trabalho de (BOLT, 1980), retirado de seu artigo original. Na figura, Richard Bolt interage com itens exibidos na tela principal da Media Room.

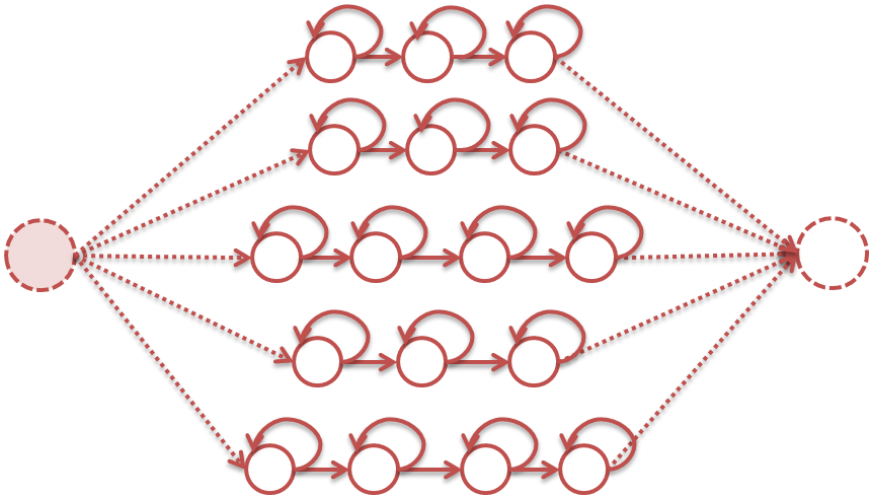

Figura 2. Representação de um classificador baseado em HMMs como uma máquina de estados finitos. Cada sequência de estados em uma linha denota um modelo HMM específico para cada gesto modelado.

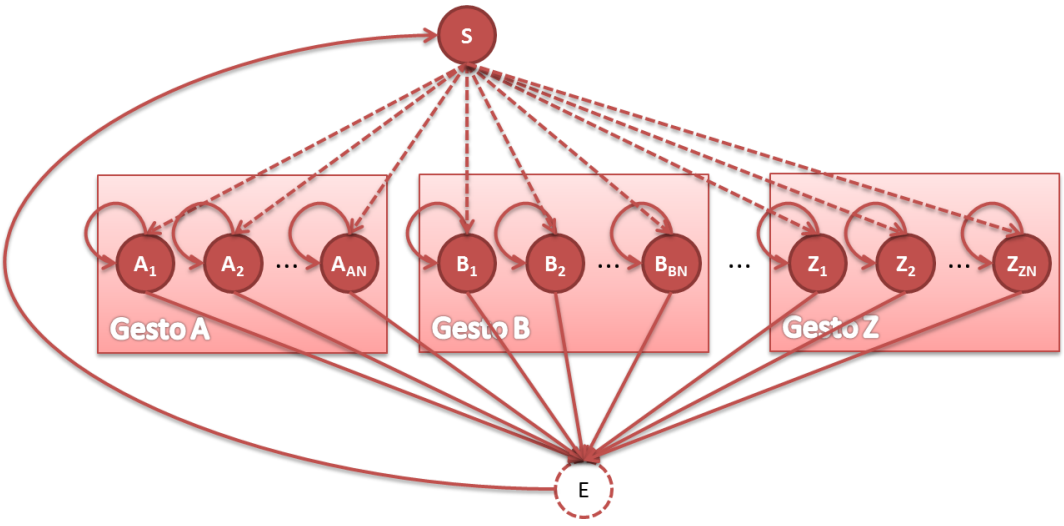

Figura 3. Exemplificação do modelo de limiar proposto por (LEE e KIM, 1999). O modelo de limiar é composto pelos estados de transição diagonal de todos os demais modelos de gestos.

Em 1999, Lee e Kim (1999) publicaram um trabalho no qual utilizavam HMMs no reconhecimento de gestos simples para o controle de uma interface visual. Uma de suas maiores contribuições foi introduzir o uso de modelos de limiar (*threshold models)* para incorporar a rejeição em classificadores baseados em HMMs. Desta maneira, em contrapartida ao trabalho de Starner, se torna possível detectar quando um sinal articulado não faz parte dos sinais esperados pelo classificador. Outra contribuição fundamental foi a criação de uma rede de detecção de gestos (*Gesture Spotting Network, GSN*) para realizar a segmentação de gestos dinâmicos conhecidos que estejam presentes em meio a um sinal temporal contínuo. Seus resultados indicaram que o uso dos modelos de limiar para realizar a rejeição alcançava até 93% de acurácia, funcionando em tempo real. Contudo, seu esquema de segmentação necessitava que um novo gesto começasse a ser articulado para que o gesto anterior fosse detectado. Para circunverter este problema, os autores utilizaram heurísticas baseadas em marcações, como a oclusão da mão ou uma parada em seu movimento.

Diferente de Starner, o trabalho de Lee e Kim considerou observações apenas discretas ao invés de contínuas, o que em parte pôde evitar problemas comumente associados à estimação de densidades Gaussianas, como o caso de variáveis constantes e matrizes de covariância não positivo-definidas. Ao invés de estimar misturas de densidades Gaussianas, Lee e Kim estimaram tabelas de frequências como funções massa de probabilidade. Como características, os autores consideraram um alfabeto discreto de 16 palavras código representando possíveis direções na movimentação da mão. A partir de dois quadros consecutivos t_0 e t_1, os autores propuseram calcular o ângulo de deslocamento θ formado entre as posições (x, y) da mão nestes dois instantes e o plano horizontal.

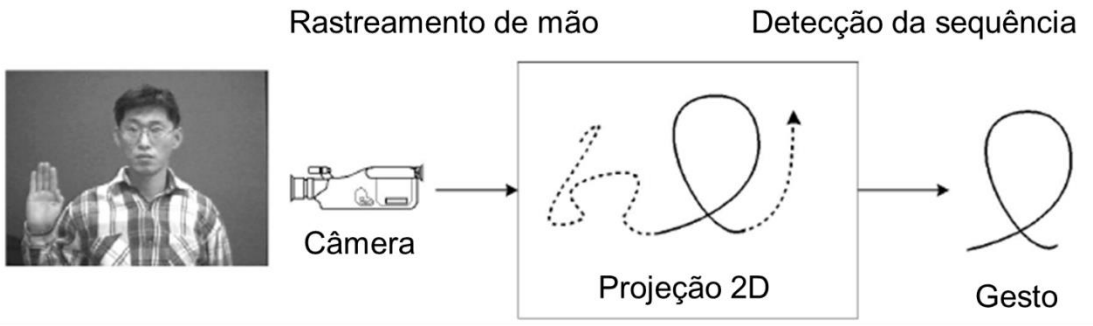

Figura 4. Sistema de rastreamento de mão e detecção de gestos criado por (LEE e KIM, 1999), imagem adaptada deste mesmo artigo.

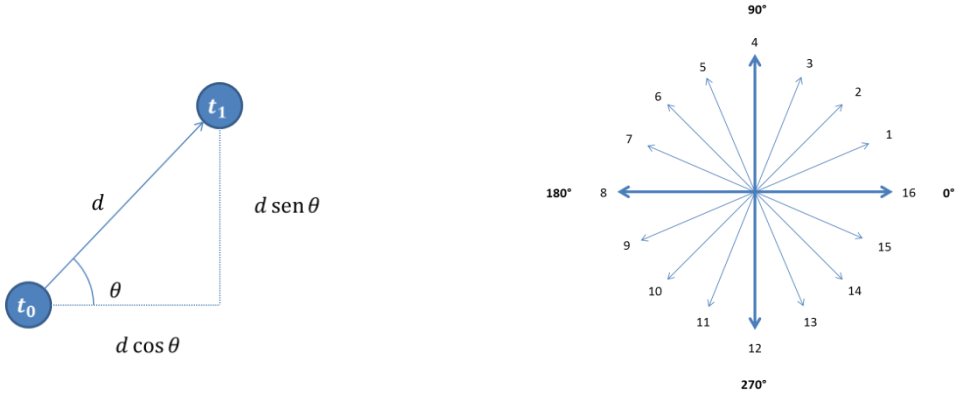

Figura 5. Discretização de trajetória baseada em palavras-código. Ângulo entre duas observações no tempo (esq.) discretização dos ângulos em palavras-código (dir.).

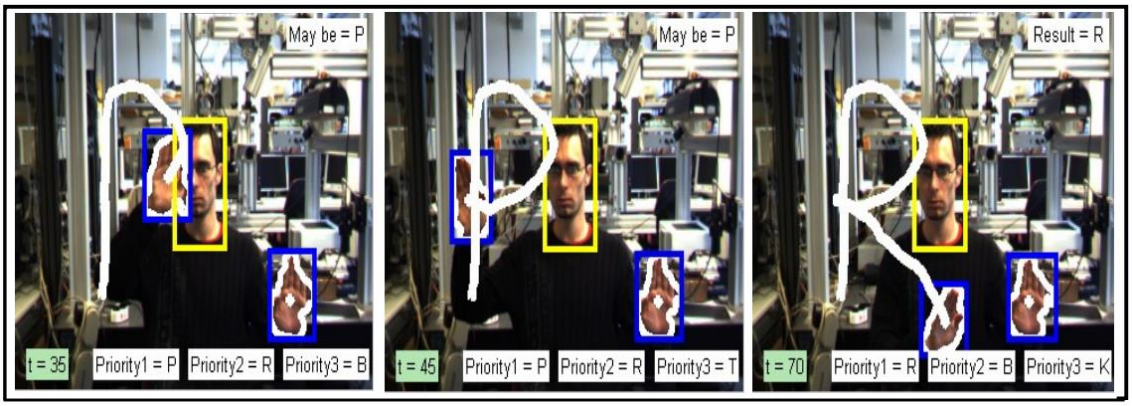

Figura 6. Imagem adaptada de (ELMEZAIN, AL-HAMADI e MICHAELIS, 2009), demonstrando a geração e os estágios de reconhecimento do gesto representando a letra R.

Em um trabalho similar ao conduzido por Lee e Kim, os autores Elmezain, Al-Hamadi e Michaelis (2009) propuseram um sistema capaz de reconhecer gestos isolados e contínuos expressando números arábicos frente a uma câmera de vídeo estéreo, em tempo real. No entanto, diferentemente do trabalho de Lee e Kim, sua abordagem utilizando visão estérea possibilitava o reconhecimento da trajetória da mão mesmo em fundos complexos. Os autores também utilizaram uma quantização em 18 símbolos, dois a mais que o trabalho de Lee e Kim, utilizando um espaçamento de 20° entre cada símbolo. Neste artigo em particular os autores utilizaram uma simples verificação baseada em velocidade estática para detectar o fim de uma sequência. Seus resultados revelam até 95.7% de acurácia em 70 sequências de vídeo contendo gestos contínuos. Em sua tese de doutorado, Elmezain (2010) também investigou o uso de Campos Aleatórios Condicionais Ocultos (*Hidden Conditional Random Fields*, HCRF) e Campos Aleatórios Latente-Dinâmicos (*Latent-Dynamical Conditional Random Fields*, LDHCRF) no lugar dos modelos ocultos de Markov

para efetuar a discriminação de sequências de observações de gestos. Nesta dissertação, campos aleatórios são discutidos com mais detalhes na seção 5.4 e 5.5.

Até este ponto, cobrimos algumas abordagens utilizando modelos estatísticos. No entanto, é possível mostrar que o tratamento de interfaces gestuais a partir de modelos alternativos também é possível. Como exemplo, podemos citar abordagens baseadas em Máquinas de Estados Finitos (*Finite State Machine*, FSM), como as utilizadas nos trabalhos de Feuerstack, Colnago e Souza (2011), Davis e Shah (1994), Bobick e Wilson (1997), e Hong, Huang e Turk (2000).

Na abordagem por FSMs, um gesto pode ser modelado como uma sequência ordenada de estados em uma configuração espaço-temporal. No trabalho apresentado em (FEUERSTACK, COLNAGO e SOUZA, 2011), seguindo uma série de estudos em interfaces gestuais inicialmente conduzidas por Feuerstack, os autores propõem uma maneira simples e intuitiva de se especificar interfaces multimodais através de diagramas de estado. Para demonstrar a aplicabilidade de sua abordagem, os autores apresentam dois estudos de caso desenvolvidos. Um dos estudos de caso apresentados consiste em uma interface natural para controlar a virada de páginas de uma partitura musical sem que seja necessária a utilização das mãos. Este é um caso atraente do uso da tecnologia de interfaces gestuais, pois resolve um problema simples, porém ativamente presente na vida de músicos, que consiste em ter de abandonar as mãos do instrumento que se está tocando para movimentar a página da partitura. No modelo de interação proposto, a movimentação da partitura é efetuada a partir do rastreio dos movimentos da face do usuário através das técnicas apresentadas nas seções 4.3 e 4.4.

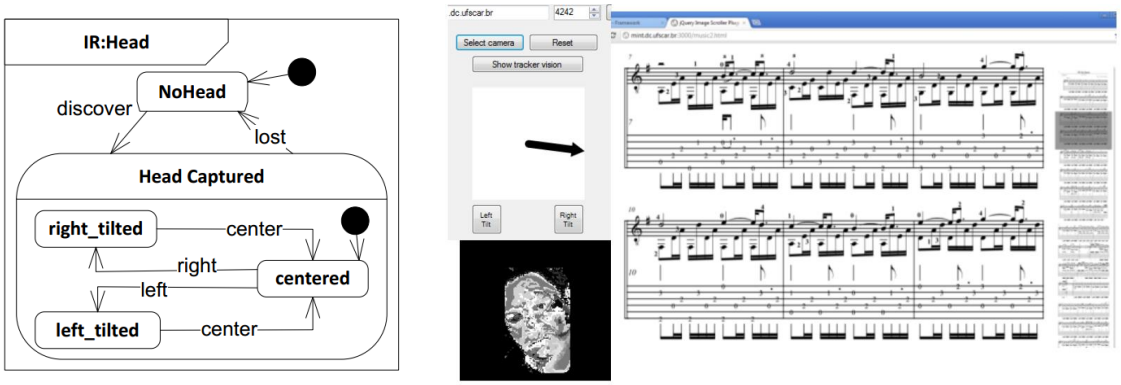

Figura 7. Controle da página atual de uma partitura através de gestos executados com a cabeça. Imagens extraídas de (FEUERSTACK, COLNAGO e SOUZA, 2011).

2.2 Reconhecimento de Línguas de Sinais

Após o passo inicial dado por Ishibuchi, Takemura e Kishino (1993) em direção ao reconhecimento de gestos através de modelos estatísticos, um dos primeiros, senão o primeiro, trabalho a utilizar HMMs no reconhecimento de Línguas de Sinais foi o desenvolvido por Starner (1995). Seu trabalho abordava o problema específico do reconhecimento da Língua de Sinais Americana (*American Sign Language*, ASL). Utilizando uma única câmera e sem utilizar uma modelagem explícita dos dedos, seu trabalho apresentava até 99.2% de acurácia. O sucesso de sua abordagem pode ser parcialmente explicado devido ao uso de um vocabulário restrito e a especificação e uso de uma gramática. Uma de suas contribuições foi justamente trazer à tona a necessidade da especificação de uma gramática para tornar o reconhecimento de frases em Línguas de Sinais tratável.

Parte da fala	Vocabulário
Pronomes	*I you he we you(pl.) they*
Verbos	*want like lose dontwant dontlike love pack hit loan*
Substantivos	*Box car book table paper pants bicycle bottle can wristwatch umbrella coat pencil shoes food magazine fish mouse pill bowl*
Adjetivos	*red brown black gray yellow*

Figura 8. Visão da câmera e vocabulário da ASL utilizado em (STARNER, 1995).

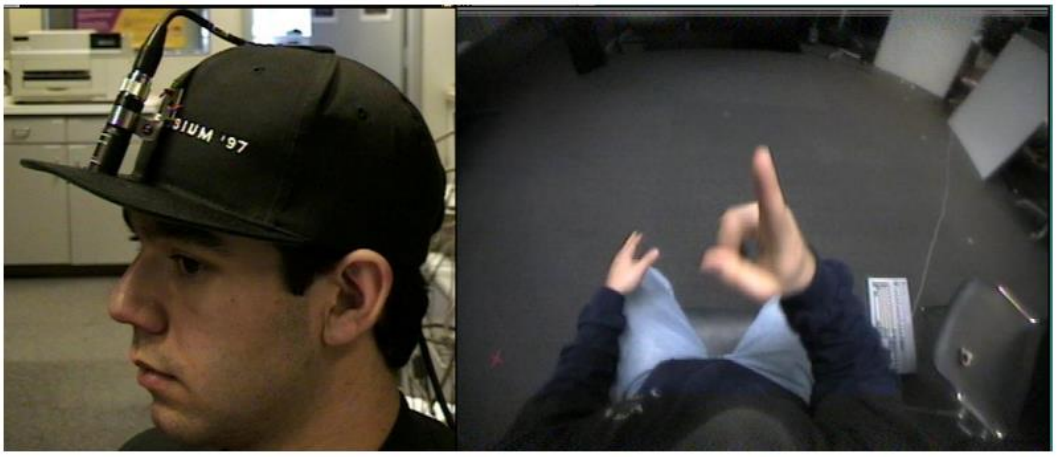

Figura 9. Extensão do trabalho de Starner para uma câmera montada sobre a cabeça do usuário, imagem extraída de (STARNER, WEAVER e PENTLAND, 1998).

Em seu trabalho, a extração de características da sequência de imagens era auxiliada com o uso de luvas coloridas, e as características extraídas eram baseadas apenas na forma e movimento (deslocamento no espaço) da mão. Após a extração da forma da mão através de segmentação de cores, o autor propôs estimar uma elipse contendo a forma da mão utilizando-se seus momentos centrais[3]. Seu vetor de características era então composto de 8 elementos: a posição (x, y), o ângulo θ do eixo de menor inércia, e excentricidade λ da elipse para cada uma das mãos. Uma vez extraídas estas características, o autor utilizou uma rede de HMMs de densidade contínua com número de estados fixos, combinados com uma gramática estatística para incorporar contexto durante o treinamento e reconhecimento. No entanto, uma desvantagem de sua abordagem é que, assim que o reconhecedor inicie o processamento da gesticulação de um usuário, este usuário poderá apenas conduzir gestos pertencentes à Língua de Sinais modelada, já que seu modelo não conseguia distinguir movimentos de mãos não previamente definidos. Seu trabalho foi continuamente desenvolvido e melhorado nos anos seguintes (STARNER, WEAVER e PENTLAND, 1998).

Além dos trabalhos para reconhecer a ASL, podemos citar também os trabalhos de Bauer e Kraiss (2002) para a Língua de Sinais Alemã (*Deutsche Gebärdensprache*, DGS), Holden, Lee e Owens (2005) para a Língua de Sinais Australiana (*Australian Sign Language*, Auslan) e Bowden *et al.* (2004), para a Língua de Sinais Britânica (*British Sign Language*, BSL). A abordagem de Bowden *et al.* (2004) se baseia na adoção de um modelo linguístico para palavras. Podemos dizer que sua abordagem é estrutural, no sentido que seu estudo tenta decompor o sinal articulado em seus visemas, da mesma maneira que uma falada pode ser decomposta em seus fonemas. Durante o primeiro estágio de segmentação e extração de características utilizando as formas limiarizadas das mãos e suas trajetórias no espaço, os atributos são convertidos em sua representação em visemas. Os visemas formadores são categorizados como as posições relativas entre as mãos, a posição das mãos em relação a pontos específicos do corpo, o movimento relativo das mãos e a sua forma. De fato, veremos no Capítulo 3 dessa dissertação que vários destes componentes são componentes da especificação do sinal na Libras e em demais Línguas de Sinais.

[3] Esta é a mesma abordagem utilizada para estimar a região da face no algoritmo de rastreio *Camshift*, apresentado e discutido com mais detalhes na seção 4.4.

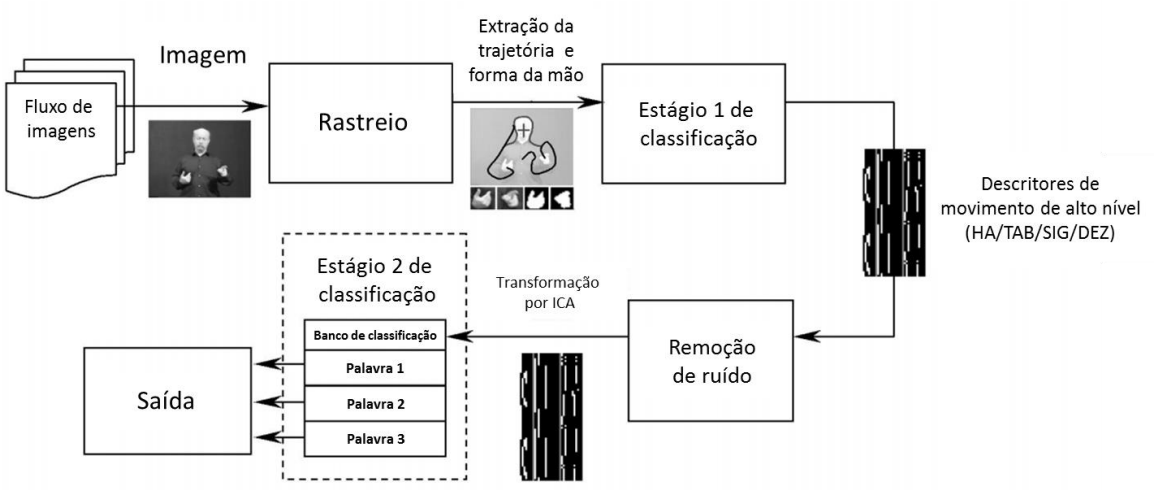

Figura 10. Diagrama em blocos exemplificando os estágios de classificação do trabalho de (BOWDEN, WINDRIDGE, et al.*, 2004), adaptado de seu mesmo artigo.*

No segundo estágio de sua arquitetura de classificação, cada sinal é modelado como uma cadeia de Markov de primeira ordem em que cada estado desta cadeia representa um conjunto particular de vetores de características obtidos do primeiro estágio de classificação. Para incorporar um modelo mais robusto das características para seu conjunto de símbolos, os autores utilizam a Análise de Componente Independente (*Independent Component Analysis*, ICA) para realizar extração de atributos. A transformação por ICA consegue separar atributos correlacionados de ruído não correlacionado, e, em efeito, acaba por reduzir a dimensionalidade do espaço de atributos. No entanto, na validação desta abordagem nesta publicação em particular, os autores utilizaram um banco de dados contendo sequências executadas por uma única pessoa. Seu vocabulário se restringia a um léxicon de 49 sinais, executados em média 5 vezes diante de uma câmera comum. Os autores utilizaram apenas um único sinal de cada palavra para treinamento, reservando todos outros para teste. Os autores reportam que o uso de ICA aumentou a taxa de classificação de 73% para 84%. Seus resultados são interessantes na medida em que exemplificam o poder de generalização dos atributos selecionados, já que a partir de um único exemplo de treinamento conseguiram obter resultados similares ao estado da arte (MITRA e ACHARYA, 2007).

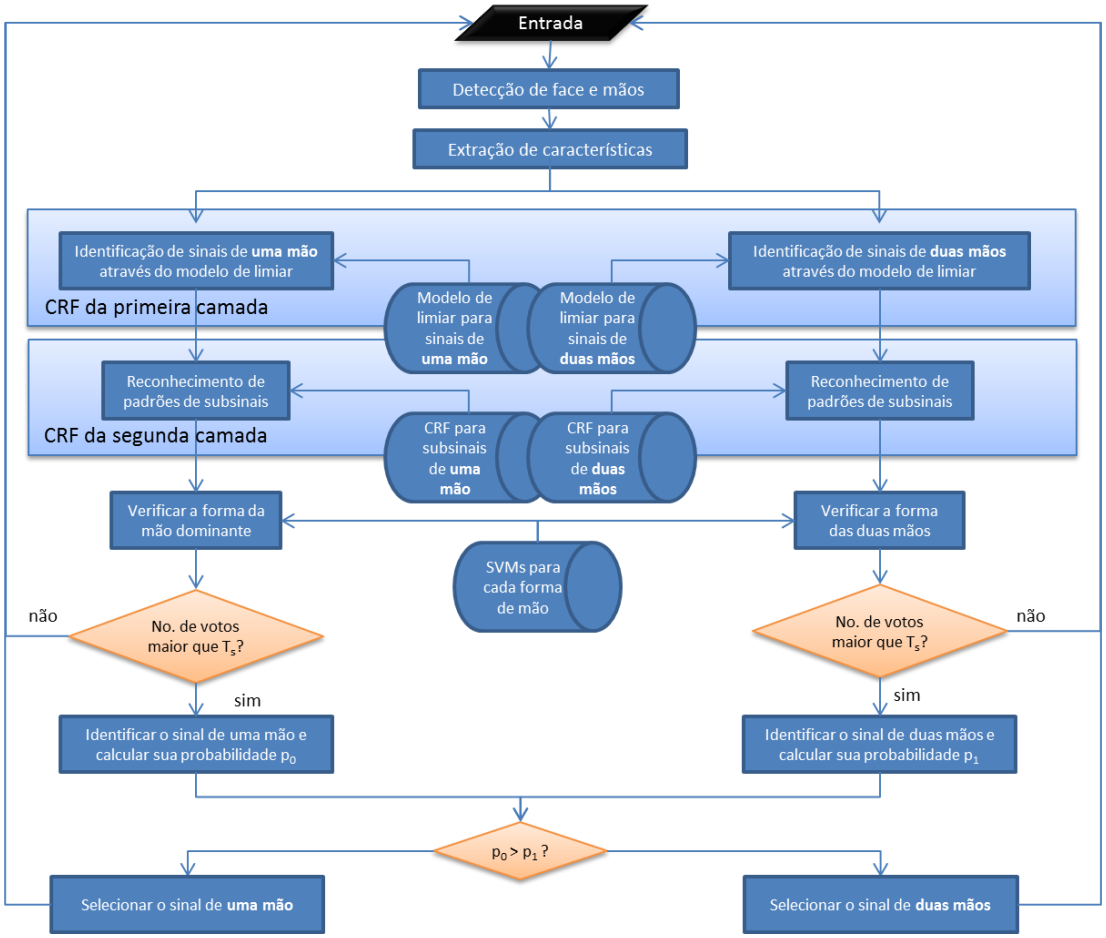

Figura 11. Diagrama de fluxo para o método proposto em (YANG, SCLAROFF e LEE, 2009). Imagem adaptada deste mesmo artigo.

Assim como Lee e Kim (1999) introduziram o conceito de modelos de limiar para realizar a identificação de gestos em sequências de padrões contínuas, Yang, Sclaroff e Lee (2009) expandiram a técnica para CRFs. Em seu trabalho, os autores propuseram um sistema capaz de realizar a distinção entre padrões de sinais e de não sinais pertencentes a um vocabulário limitado de palavras da língua de sinais estudada. Para este efeito, os autores estenderam a técnica de Lee e Kim para criar um modelo de rejeição a partir de um CRF. Em seu sistema, cada estado do CRF se refere a um rótulo de um sinal em particular da Língua de Sinais modelada. Os autores propõem então incorporar um rótulo adicional representando um não-sinal, cujos pesos de ocupação e transição são então inicializados de acordo com seu método proposto. Assim como no trabalho de Lee e Kim, os autores utilizam este modelo de limiar para realizar a identificação de um sinal através de um algoritmo de detecção de final de sua articulação. Seu método é então incorporado como

componente de um sistema maior utilizado para a detecção, reconhecimento e filtragem dos sinais detectados, organizado em uma arquitetura de duas camadas.

O trabalho de Pizzolato, Anjo e Pedroso (2010) concentrou-se sobre o problema de detecção de gestos estáticos e dinâmicos em Libras, e, assim como o trabalho anterior, também utilizou uma arquitetura em duas camadas. No entanto, seus dois estágios de classificação consistiam em ANNs organizadas de maneira que símbolos notoriamente distintos do alfabeto manual da Libras pudessem ser detectados mais rapidamente, e símbolos que sabidamente resultariam em maior confusão aos classificadores pudessem ser refinados e fornecidos a classificadores mais especializados.

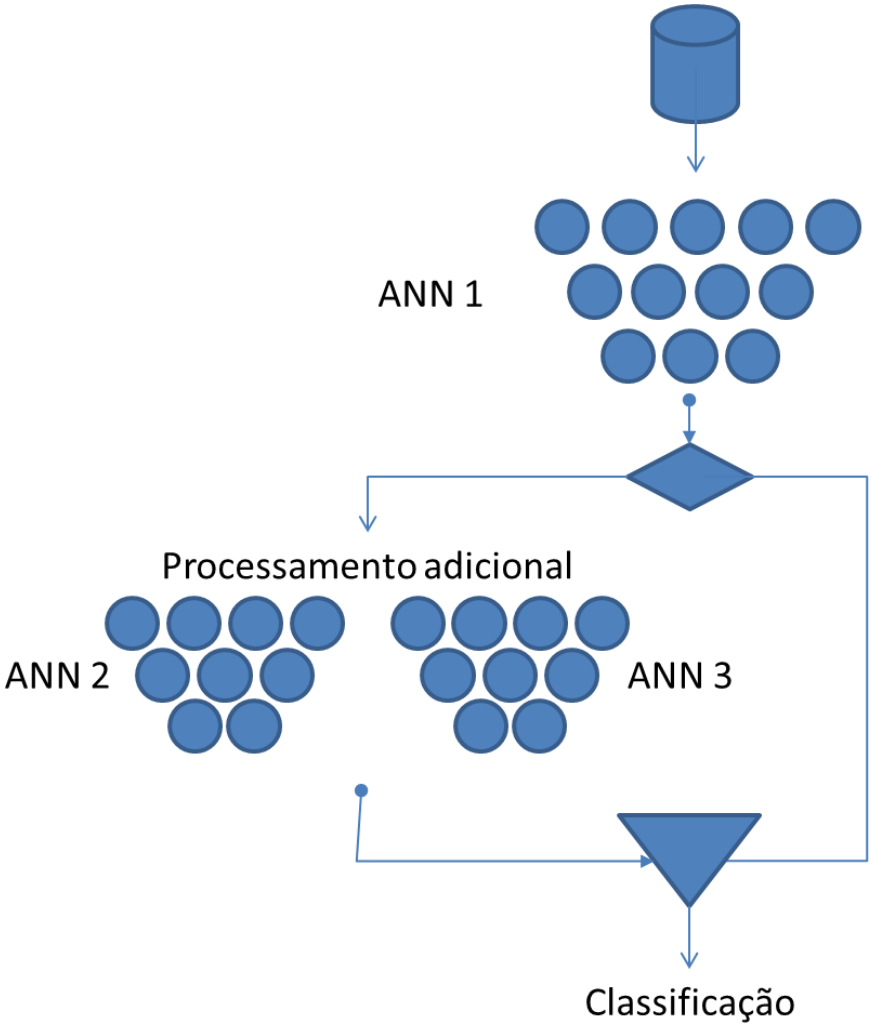

Figura 12. Arquitetura em duas camadas para detecção de gestos estáticos propostas em (PIZZOLATO, ANJO e PEDROSO, 2010).

Cada camada responsável pelo reconhecimento de poses utilizava classificadores baseados em redes neurais para realizar a identificação da postura sendo realizada. Após a postura ser detectada, sua identificação era então informada a um algoritmo de detecção de gestos dinâmicos responsável por realizar a classificação da sequência de posturas detectadas e então lhe atribuir um rótulo correspondente a esta palavra. Os modelos de classificação de sequências eram, assim como nos trabalhos de Starner (1995), baseados em HMMs. No entanto, o trabalho destes autores se dedicava a classificar palavras soletradas da Libras a partir de seu alfabeto manual. Os autores criaram seu próprio banco de dados para a Libras, contendo 27 sinais, 19 gestos estáticos (poses) e 8 gestos dinâmicos. Um total de 45 pessoas foram selecionadas para criar o banco de dados. Os autores optaram por utilizar um algoritmo de segmentação baseado em cores, e, para tanto, os participantes utilizaram jaquetas que casassem com a cor do fundo, de forma que apenas a mão ficasse aparente na sequência de imagens. Nesta dissertação, o alfabeto manual da Libras é apresentado e discutido com maiores detalhes na seção 3.5.

Enquanto o trabalho de Pizzolato, Anjo e Pedroso (2010) se dedicou ao estudo de sinais da Libras formados por componentes estáticos, o trabalho de Dias *et al.* (2009) se concentrou em estudar especificamente os tipos de movimentos presentes na língua. Como será apresentado na seção 3.2 desta dissertação, veremos que o movimento faz parte da estrutura básica dos gestos de Línguas de Sinais como a Libras. Em seu trabalho, os autores investigaram o uso de mapas auto-organizáveis (*Self-Organizing Maps*, SOM) e modelos nebulosos de aprendizado de quantização vetorial (*Learning Vector Quantization*, LVQ) para o reconhecimento de diferentes tipos de movimentos presentes na Libras. Estes diferentes tipos de movimentos são apresentados com mais detalhes na seção 3.2.3. Os autores especificaram o problema do reconhecimento de gestos na Libras representando os movimentos das mãos ao decorrer do tempo como curvas bidimensionais em $\mathbb{R}^2$. Seu método de segmentação é baseado em cor, sendo que para auxiliar a segmentação, os usuários tiveram de utilizar luvas coloridas.

Outro trabalho envolvendo o uso de SOM foi o trabalho conduzido por Carneiro, Cortez e Costa (2009), que as empregou como etapa de pré-classificação dos momentos invariantes de Hu (HU, 1962) das imagens de gestos. O autor reporta taxas de reconhecimento superiores as 98% para o reconhecimento dos 26 gestos do alfabeto manual da Libras.

2.3 Reconhecimento de Expressões Faciais

A expressão facial possui um papel importante na expressão de informação em línguas gestuais, e na Libras não é exceção. No entanto, existem severos fatores complicadores envolvidos em seu reconhecimento. Humanos são capazes de detectar e identificar faces em uma cena com mínimo de esforço, ainda que estas faces estejam sujeitas a variações de luminosidade, variações entre gêneros, variações devido à idade, oclusão, óculos, estilos de cabelo ou disfarces (MITRA e ACHARYA, 2007). Abordagens incluem o uso de ANNs, PCA, LDA, SVD, fluxo óptico (*optical flow*) e filtros de partículas (*particle filters*).

Em um exemplo clássico na literatura de reconhecimento de faces, Rowley, Baluja e Kanade (1998) utilizaram uma ANN para examinar pequenas janelas de 20×20 pixels a fim de decidirem se estas janelas continham ou não uma face. Esta abordagem requeria que a imagem fosse continuamente redimensionada e a janela deslocada até que toda a imagem tivesse sido analisada. Seu desempenho ainda não era compatível com aplicações em tempo real. Um grande passo em direção ao reconhecimento de faces em tempo real foi dado a partir da publicação de Viola e Jones (2001). Em seu trabalho, os autores propuseram um *framework* geral de reconhecimento de objetos com particular aplicação à detecção de faces capaz de operar em tempo real em computadores comuns de sua época. O feito foi conseguido através de várias inovações trazidas pelos autores, como o uso de imagens integrais e técnicas de *boosting*. Estas inovações e o método como um todo é descrito com maiores detalhes na seção 4.3 dessa dissertação.

Quanto ao reconhecimento de expressões faciais, podemos destacar o trabalho de Bartlett *et al.* (2003). Seu trabalho apresenta uma das abordagens de maior sucesso no reconhecimento de expressões faciais, baseando-se no uso de características extraídas de filtros de Gabor posteriormente classificadas através de SVMs. No entanto, Whitehill e Omlin (2006) mostraram que, apesar do sucesso desta abordagem, seu custo computacional em termos de memória e tempo de processamento é bastante alto. Sua investigação mostrou que o uso de características de *Haar* em classificadores treinados através de algoritmos de *boosting* é capaz de proporcionar taxas de sucesso similares, porém operando de maneira mais rápida *por várias ordens de magnitude*.

2.4 Segmentação e representação de imagens

Para dar suporte a detecção e reconhecimento de gestos, é imprescindível que primeiro exista uma adequada seleção e representação de atributos. Para tratar este problema, grande parte da pesquisa em Processamento de Imagens e Sinais tem se dedicado ao problema particular de detecção, identificação e segmentação de humanos em imagens e sequências de imagens.

Como enunciam Schwartz *et al.* (2009), duas abordagens principais tem sido seguidas nos últimos anos para se obter a segmentação e identificação do corpo humano em imagens. A primeira delas consiste em uma abordagem gerativa, em que partes do corpo humano são detectadas separadamente, e então combinadas de acordo com algum modelo *a priori*. A segunda classe de métodos consiste em empregar a análise estatística pura a conjuntos de características extraídas a partir de janelas da imagem, e analisar estas características como um todo de maneira a detectar se há o objeto de interesse dentro desta janela ou não.

Figura 13. Detecção de pedestres utilizando PLS, retirada de (SCHWARTZ, KEMBHAVI, et al.*, 2009) e reproduzida com permissão do autor.*

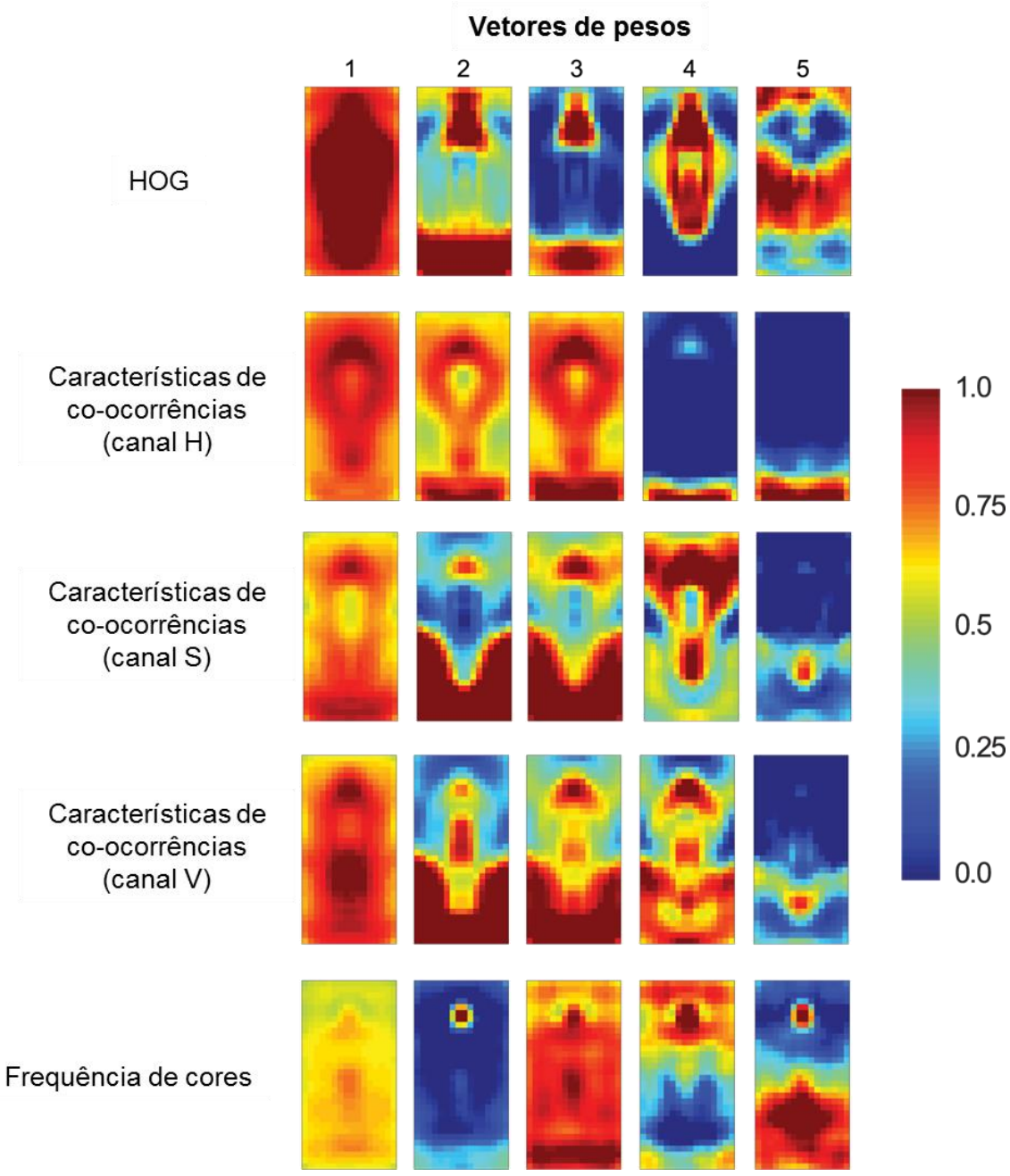

Figura 14. Características utilizadas para reconhecimento de pedestres em (SCHWARTZ, KEMBHAVI, et al.*, 2009), imagem adaptada com permissão do autor.*

Um dos algoritmos mais conhecidos para se realizar a detecção de objetos é o algoritmo de Viola e Jones (2001), também conhecido como classificador em cascata de *Haar*. Este método é um método geral de detecção de objetos e pode ser treinado para reconhecer diversos objetos de interesse. Sua abordagem em si pertence à segunda classe de métodos referidos por Schwartz e colegas (2009). Porém, por tratar-se de um método de detecção geral, pode ser utilizado para detectar partes do corpo em separado para que possam ser mais tarde combinadas em um modelo único como os da primeira classe antes mencionada. O método de Viola-Jones é apresentado e discutido em detalhes na seção 4.3 desta dissertação.

O método de Viola-Jones é um método de detecção baseado no janelamento de uma cena. Métodos baseados em janelamento podem extrair diversas características de uma determinada janela, ou região da imagem, e efetuar a classificação destas características baseadas em qualquer técnica de reconhecimento de padrões. Como tese de doutorado, Schwartz propôs o uso da técnica de Mínimos Quadrados Parciais (*Partial Least Squares*, PLS) como maneira de circunverter o problema da alta dimensionalidade do espaço de classificação tipicamente enfrentado por algoritmos de detecção de imagens baseados em janelas (SCHWARTZ, KEMBHAVI, *et al.*, 2009). A técnica de PLS é comumente utilizada em quimiometria para buscar uma representação adequada de menor dimensionalidade em aplicações de espectrometria (WOLD, SJÖSTRÖM e ERIKSSON, 2001), e até então sua aplicabilidade não havia sido estudada no contexto da visão computacional.

Em seu trabalho, Schwartz explora esta capacidade da análise de manipular dados de alta dimensionalidade para aplica-la em janelas contendo 170,820 características extraídas a partir de descritores baseados em histogramas de gradientes orientados (*Histograms of Oriented Gradients*, HOG), coocorrências extraídas de canais de cores num espaço HSV e frequência de cores. Estas características visam explorar mais informações, como a homogeneidade de roupas, cores, e texturas, do que outros métodos propostos em sua área.

O trabalho de Schwartz apresenta resultados tão bons ou melhores do que outras técnicas do estado-da-arte, como SVMs, porém a um custo computacional significantemente menor. Seu método mostra ser superior a outros métodos do estado-da-arte em três conjuntos populares de dados, como o conjunto INRIA de imagens de pedestres. O uso de HOGs em (DALAL e TRIGGS, 2005) também se mostrou superior a diversas outras características como as baseadas em Wavelets (MOHAN, PAPAGEORGIOU e POGGIO, 2001).

Ghobadi e colegas (GHOBADI, LOEPPRICH, *et al.*, 2007) investigaram o uso de imagens de profundidade para auxiliar na tarefa de segmentação das mãos. Sua técnica baseia-se na fusão das características de profundidade e de intensidade para aumentar a acurácia da segmentação em imagens capturadas através de uma câmera de tempo-de-voo[4]. Sua técnica de fusão de características baseia-se na mesma técnica utilizada em outras áreas como no processamento e análise de imagens de radares de abertura sintética (*Synthetic Aperture Radar*, SAR).

Os resultados deste estudo mostraram que imagens geradas apenas a partir da profundidade ocasionam muitos erros de segmentação, e que sua abordagem de fusão das características superam muitas destas dificuldades. O fato de que não é possível colocar tamanha expectativa em métodos de classificação baseando-se apenas em imagens de profundidade também é advertido por Helmer e Lowe (2010) em seu trabalho sobre reconhecimento de objetos em visão estéreo.

Porém em 2011, Shotton *et al.*, pesquisadores da divisão de pesquisas da Microsoft, publicaram um artigo sobre a segmentação e reconhecimento da pose humana utilizada na plataforma de jogos *Kinect*. O método descrito é capaz, de maneira eficaz e veloz, de predizer as posições 3D de junções do corpo humano de uma única imagem de profundidade, sem que seja necessário utilizar rastreamento. A detecção e segmentação das partes do corpo são feitas a cada quadro, de maneira que partes do corpo sejam detectadas numa imagem independente de informação temporal (SHOTTON, FITZGIBBON, *et al.*, 2011).

A abordagem utilizada segue a de reconhecimento de objetos de maneira que o problema possa ser modelado como um simples problema de classificação *pixel-a-pixel*. Por utilizar um imenso conjunto de dados de treinamento, compreendendo tanto dados reais quanto sintéticos, seu classificador é capaz de gerar predições independentes de pose, forma do corpo e roupas. O sistema roda a 200 quadros por segundo e mostra grande acurácia tanto nos conjuntos sintéticos quanto reais. Mais detalhes sobre o sensor *Kinect* e mapas de profundidade serão apresentados na seção 4.1 desta dissertação. Uma prévia de diferentes imagens de profundidade conforme apresentadas no artigo original é apresentada na Figura 15.

Um ponto interessante que deve ser levado em consideração é o fato de que o sensor *Kinect* funciona mesmo sem a plataforma de interação da Microsoft. Como exemplo, temos o trabalho de Oikonomidis, Kyriazis e Argyros (2011), em que os

[4] Uma câmera de tempo-de-voo realiza a medição de distâncias em uma cena tendo como base o tempo que um raio de luz que deixa a câmera consome para chegar a um objeto da cena, refletir, e atingir a câmera novamente.

autores criaram um método para rastreio das articulações da mão através da minimização da divergência entre o que está sendo observado e um modelo de junções tridimensionais utilizando uma variação do método de Otimização por Nuvem de Partículas (*Particle Swarm Optimization*, PSO) em uma GPU. Segundo os autores, esta técnica é capaz de realizar o reconhecimento e o rastreio das partes da mão (exemplificado na Figura 16) a uma taxa de até 15Hz, sendo um dos métodos mais robustos e acurados para este fim baseado em modelos 3D.

Figura 15. Exemplo de poses em imagens de profundidade geradas por um sensor Kinect. Adaptado de (SHOTTON, FITZGIBBON, et al.*, 2011).*

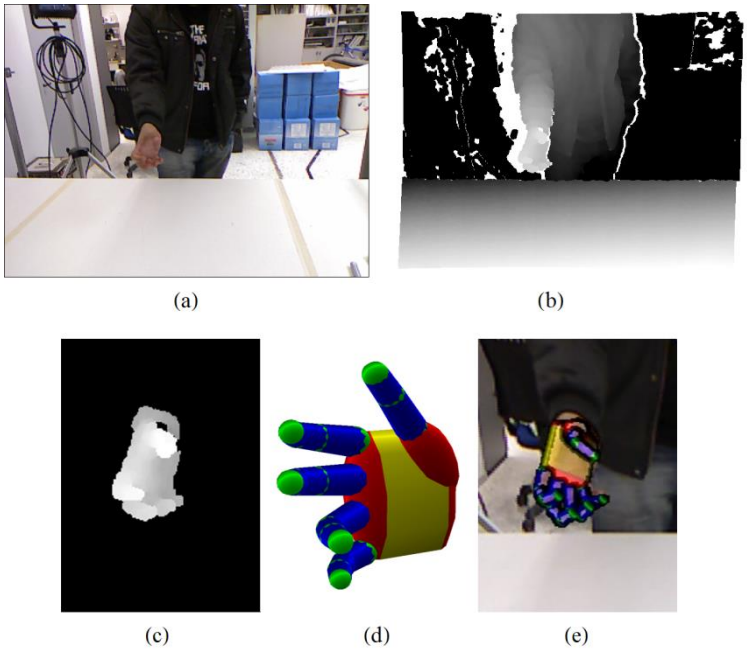

Figura 16. Figura adaptada de (OIKONOMIDIS, KYRIAZIS e ARGYROS, 2011) exibindo sua abordagem baseada em modelos. As figuras (a) e (b) exibem as imagens coloridas e de profundidade capturadas pelo sensor Kinect, respectivamente. A imagem (c) exibe a segmentação da mão, (d) mostra o modelo de articulações utilizado, e (c) a estimação deste modelo para a cena apresentada.

2.5 Resumo do capítulo

Neste capítulo, apresentamos como sistemas de reconhecimento de gestos têm sido desenvolvidos na literatura, dando ênfase especial a modelos baseados em visão computacional. Aqui, vimos como muitos trabalhos baseiam-se nos mesmos mecanismos utilizados para a criação de sistemas de reconhecimento de fala, e como tal, aproveitam-se dos mesmos modelos estatísticos populares nesta área.

Tal como no reconhecimento de fala, em que palavras faladas são divididas em suas unidades elementares – os fonemas – observamos que algumas abordagens para reconhecimento de Línguas de Sinais também particionam o sinal em suas unidades básicas – os visemas. Sendo esta característica fundamental para a criação de sistemas de reconhecimento de fala, nos próximos capítulos exploraremos a estrutura formadora do sinal gestual e buscaremos meios de usar esta estrutura a nosso favor na elaboração de métodos para reconhecimento da Libras.

Capítulo 3

A Língua Brasileira de Sinais

"A língua é a chave para o coração de um povo. Se perdemos a chave, perdemos o povo. Se guardamos a chave em lugar seguro, como um tesouro, abriremos as portas para riquezas incalculáveis, riquezas que jamais poderiam ser imaginadas do outro lado da porta." — Eva Engholm, 1965

A Língua Brasileira de Sinais foi reconhecida como língua oficial no país em 2002 após a instauração da lei n° 10.436 de 24 de abril de 2002 e o decreto n° 5626 de 22 de dezembro de 2005 que a regulamenta. Em seu parágrafo único, este decreto define por Língua Brasileira de Sinais "a forma de comunicação e expressão, em que o sistema linguístico de natureza visual-motora, com estrutura gramatical própria, constitui um sistema linguístico de transmissão de ideias e fatos" (BRASIL, 2002) originada nas comunidades surdas do país. Neste capítulo, apresentaremos as principais características da Libras, bem como o papel que desempenha na vida do surdo e as dificuldades envolvidas em seu uso e aprendizado.

3.1 Introdução

Estendendo a definição oficial, podemos dizer que a Língua Brasileira de Sinais apresenta não apenas estrutura gramatical própria, mas traz também todo um arcabouço léxico e semântico próprio. Como as demais Línguas de Sinais, atende todos os critérios linguísticos de uma língua genuína, incluindo sua capacidade de gerar uma quantidade infinita de sentenças[5]. Como bem observam Quadros e Karnopp, a Língua Brasileira de Sinais é considerada pela linguística uma língua natu-

[5] Observação constatada por Willian Stokoe durante seus estudos da Língua de Sinais Americana, conforme contada em (O'BRIEN, 2005, p. 101).

ral e não apenas "um problema do surdo ou uma patologia da linguagem" (QUADROS e KARNOPP, 2004).

No Brasil, o reconhecimento da Libras como língua oficial também garantiu ao surdo o direito ao intérprete no que diz respeito à área penal (como em depoimentos e julgamentos de surdos) e no processo de inclusão social na educação. O papel do intérprete se torna fundamental na inclusão social do surdo, e como tal, tem levado a estudos sobre o estado atual desta profissão (GUARINELLO, SANTANA, *et al.*, 2008; ROSA, 2003; ROSA, 2008) e das dificuldades que estes intérpretes encontram (FAMULARO, 1999; NICOLOSO e SILVA, 2009).

Uma das dificuldades imediatas da profissão de intérprete é sua falta de regulamentação. Em uma pesquisa conduzida por Guarinello *et al.* (2008) com estudantes universitários surdos, a maioria dos entrevistados se considerou mais proficiente na Língua de Sinais do que na modalidade oral do português. No entanto, como bem observa estes autores, esta autoavaliação da proficiência é extremamente subjetiva. Ao considerar que muitos surdos só tiveram acesso a Libras na adolescência, os autores observam que não há informação sobre os interlocutores com quem aprenderam a Língua de Sinais. Os autores argumentam que, como há muitos *maus falantes* de Libras no país, que utilizam sinais de forma gramaticalmente incorreta e fazendo uso do bimodalismo e da comunicação total[6], seja razoável concluir que alunos que tenham aprendido com estes interlocutores também se tornem *maus falantes* da língua de sinais (e que há uma gama de variáveis com implicações diretas no que pode se entender como proficiência de uma língua).

A interpretação de Libras não é uma tarefa fácil. Como relatam Nicoloso e Silva (2009), na Libras pode ocorrer o caso de um mesmo sinal, articulado da mesma maneira, pertencer a classes gramaticais diferentes conforme o contexto da frase. Não apenas isto, mas em seu trabalho as autoras focam no problema da identificação do sujeito nas narrativas em Libras. Na Língua Brasileira de Sinais, é possível armazenar o sujeito[7] em um local do espaço e retomá-lo a qualquer momento durante o discurso através do uso da apontação. É necessário ao intérprete, portanto, extrema atenção e capacidade de memória para que consiga distinguir entre os pontos do espaço que armazenam cada um dos nominais abordados na narrativa. En-

[6] A comunicação total defende a utilização de todos os recursos linguísticos, como quaisquer resquícios de habilidade de fala, audição, leitura facial e línguas gestuais de maneira concomitante na comunicação.

[7] O sujeito aqui é referido tanto em seu sentido gramatical quanto no sentido daquilo de que se fala ou a que se atribuem qualidades e designações durante o discurso.

quanto que na língua portuguesa é possível inferir de quem se fala através de indicações subjetivas (certas "pistas", como enunciam as autoras), na Libras esta atividade se torna mais difícil. Por fim, vale ressaltar que a Libras é também uma língua viva, e como tal, se renova e se modifica a cada dia. Assim como a língua falada, possui regionalismos, dialetos e gírias. Isto dificulta ainda mais sua interpretação, visto que palavras podem ser representadas por diferentes sinais em diferentes regiões do país.

3.2 Estrutura interna dos sinais

Como toda língua natural, a Libras possui estrutura gramatical e sintaxe próprias. A estrutura básica da Libras é constituída de parâmetros primários e secundários que se combinam de forma sequencial ou simultânea. Isto significa que cada gesto pode ser expresso através de uma combinação destes parâmetros, assim como palavras faladas são compostas de fonemas.

Segundo Ferreira Brito (2010, p. 36-41) os parâmetros primários são a *configuração das mãos (CM)*, os *pontos de articulação (PA)*, e o *movimento* (M). Os parâmetros secundários são: a *disposição das mãos (DM) orientação das palmas das mãos (OM)*, a *região de contato* (RC) e as *expressões faciais* ou *componentes não manuais* (NM). Alguns dos parâmetros mencionados adiante, em especial as configurações das mãos e as expressões faciais, podem ser considerados morfemas da Língua de Sinais e não apenas fonemas. Adotando esta parametrização do sinal, seria possível dizer, em resumo, que o sinal na Libras pode ser representado pela tupla

$$S = (CM, PA, M, DM, OM, RC, NM)$$

em que $CM \in \Lambda$, sendo Λ o conjunto das 46 possíveis configurações das mãos; $PA \in \Omega$, sendo Ω o conjunto de possíveis regiões no espaço de sinalização; e $M = (d, u, t)$ com $d \in D$, $\mathrm{u} \in U$, e $t \in T$ denotam as características do movimento a ser realizado. Para o componente de movimento, D denota o conjunto das possíveis *direções* de movimento, U denota o conjunto dos possíveis *tipos* de movimento, e T o conjunto de possíveis *intensidades* destes movimentos. Continuando com a definição dos demais componentes, $OM \in P$, sendo P o conjunto de possíveis direções da palma da mão; $RC \in C$, em que C é o conjunto de possíveis tipos de toque; e

$NM \in E$, em que E é o conjunto de possíveis expressões não manuais. Finalmente, $DM \in \{dominante, ambas\}$, designando quais mãos devem executar o gesto.

Nas subseções seguintes, discursaremos sobre o significado de cada um destes parâmetros e apresentaremos sugestões para a especificação dos conjuntos Λ, Ω, D, U, P, C e E acima apresentados. É importante ressaltar, no entanto, que há divergências entre autores quanto a completa especificação destes conjuntos, e o seguinte servirá apenas para ilustrar estas possíveis caracterizações do sinal na Libras.

3.2.1 Configuração das mãos (CM)

As possíveis configurações das mãos sumarizam as possíveis disposições dos dedos em um conjunto finito de símbolos. De acordo com Ferreira Brito (2010), são 46 as configurações de mãos presentes na Libras[8]. A figura exibida na página seguinte, extraída do trabalho ante citado, visa ilustrar cada uma destas possíveis configurações.

As configurações de mãos desempenham papel fundamental como classificadores na Libras. Classificadores são morfemas que, anexados a outras palavras, estabelecem marcas de concordância de gênero. Como exemplo, podemos citar que a sinalização do verbo ANDAR referindo-se a uma pessoa é diferente da sinalização do verbo CAIR quando se referindo a um animal. Outra observação interessante pode ser feita a respeito das configurações de mãos: Conforme ilustrado na Figura 17, várias configurações são também elementos da datilologia da Libras (Figura 25). Isto é um indicativo que quaisquer esforços direcionados a detecção automática do alfabeto manual da Libras também se torna automaticamente aplicável ao reconhecimento de um subconjunto específico das possíveis configurações de mãos.

[8] Parece haver certa divergência entre autores. O dicionário digital de (LIRA e SOUZA, 2008) cita 73 destas possíveis configurações. No entanto, neste trabalho, aderiremos à definição dada por Ferreira-Brito.

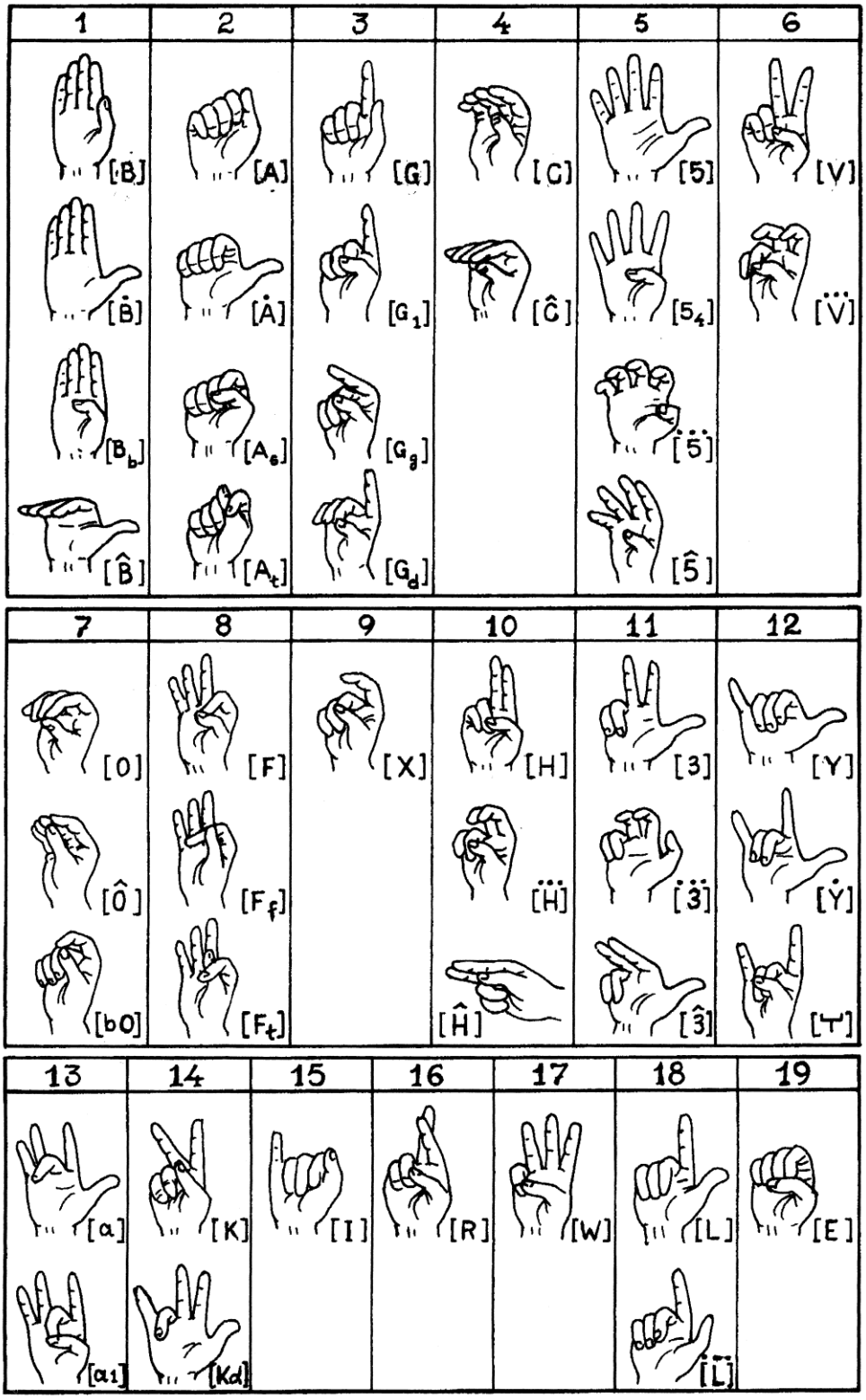

Figura 17. Configurações das mãos presentes na estrutura da Libras, extraído de (FERREIRA-BRITO, 2010).

3.2.2 Pontos de articulação (PA)

Os pontos de articulação na Libras designam as possíveis regiões do espaço em relação ao corpo onde os sinais são articulados (FERREIRA-BRITO, 2010). Podemos, por exemplo, ter um sinal articulado junto à testa, ou mesmo um sinal articulado junto às pernas. Para caracterizar cada uma destas possibilidades, o gesto na Libras é também parametrizado em relação a estas possíveis regiões. A região do espaço imediatamente a frente do interlocutor é denominada espaço neutro. A caracterização do espaço disponível a articulação dos sinais é apresentada na Figura 18, e a lista dos pontos de articulação apresentadas por Brito é mostrada na Figura 19.

3.2.3 Movimento (M)

O movimento executado no sinal da Libras denota o deslocamento no espaço efetuado pelas mãos durante a sinalização. Como nota Ferreira Brito (2010, p. 74), na Libras sempre há pelo menos um eixo fixo durante o movimento, e há poucas possibilidades para a posição deste eixo. Quando as rotações em torno deste eixo são significativas, estas rotações em geral caracterizam meio-círculos ou um quarto de círculo. Os graus de liberdade para o movimento das mãos são ilustrados na Figura 22.

Segundo Strobel e Fernandes (1998), o movimento pode ser caracterizado em termos de sua direcionalidade e seu tipo de movimento. A direcionalidade do movimento caracteriza a direção em que as mãos se movem durante a execução do sinal. Um movimento unidirecional envolve o movimento de apenas uma das mãos na direção especificada. Um movimento bidirecional envolve o movimento de uma ou ambas as mãos em direções diferentes. Já um movimento multidirecional explora diversas direções durante sua execução. O tipo do movimento caracteriza a forma de sua execução. As autoras Strobel e Fernandes enumeram seis tipos de movimentos existentes na Libras, ilustrados na Figura 20.

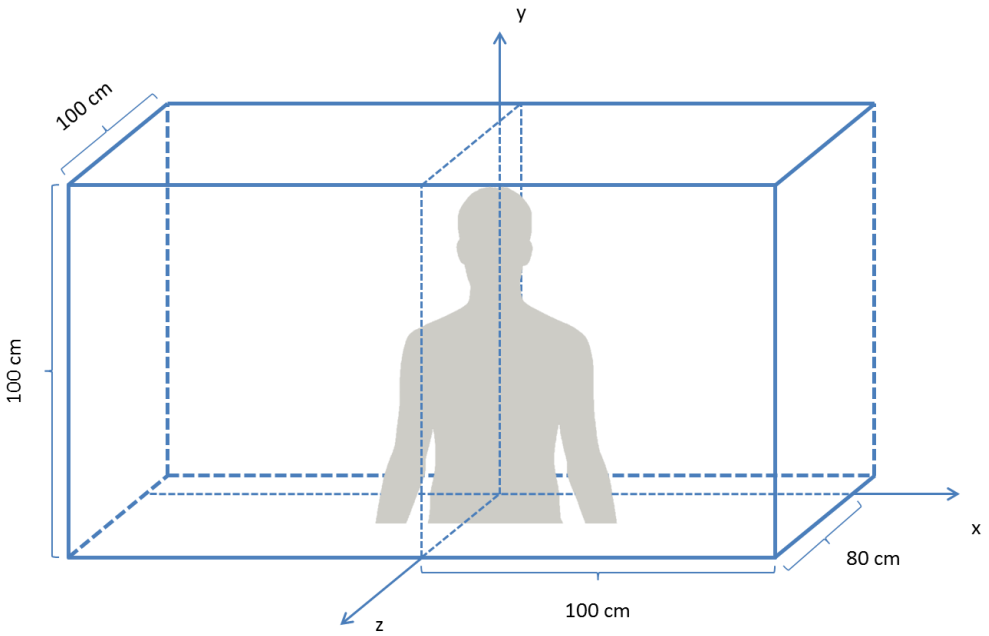

Figura 18. O espaço de sinalização. Imagem recriada tendo como base o trabalho de (FERREIRA-BRITO e LANGEVIN, 1994).

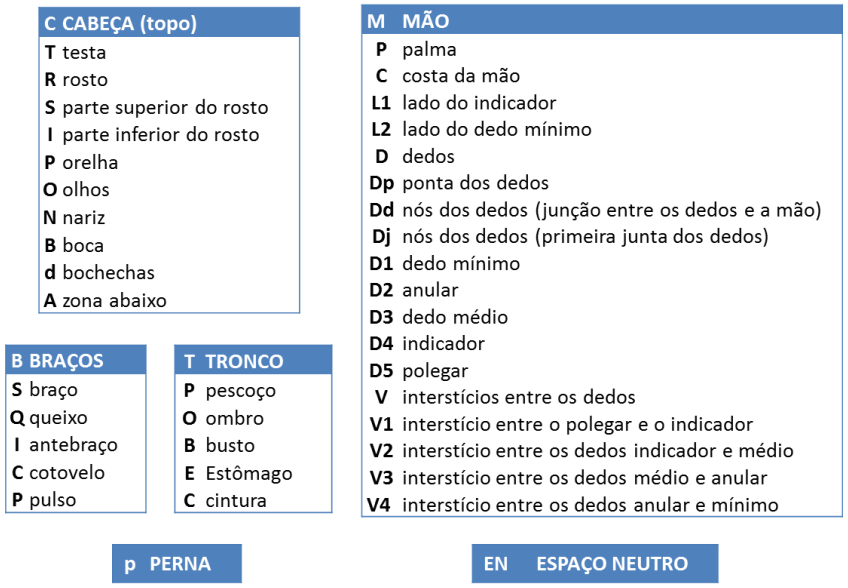

Figura 19. Pontos de articulação da Libras segundo (FERREIRA-BRITO, 2010).

Outro aspecto importante do movimento é sua velocidade e intensidade. Para caracterizar estes aspectos, Ferreira Brito (2010, p. 228) propõe a definição de quatro componentes do movimento dependentes da velocidade de execução, sendo eles a tensão, retenção, continuidade e refreamento, cujas características são representadas na Figura 21. Para finalizar este tópico, poderíamos dizer que o movimento pode ser caracterizado pela sua direcionalidade $d \in D$, tipo $\mathrm{u} \in U$ e intensidade $t \in T$, conforme especificados pelos conjuntos a seguir.

$$D = \{\, \text{Unidirecional, Bidirecional, Multidirecional} \,\}$$

$$U = \{\, \text{Retilíneo, Helicoidal, Circular, Semicircular, Sinuoso, Angular} \,\}$$

$$T = \{\, \textit{Tenso, Com Retenção, Contínuo, Refreado} \,\}$$

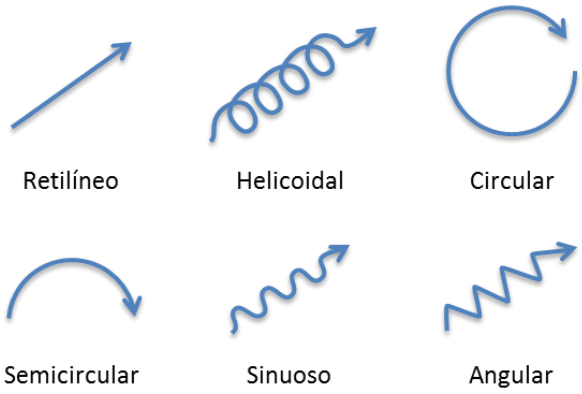

Figura 20. Tipos de movimentos em Libras

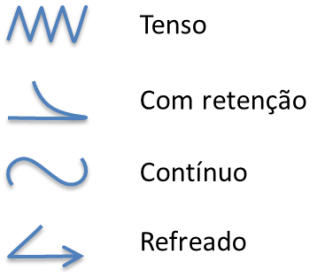

Figura 21. Componentes do movimento dependentes da velocidade.

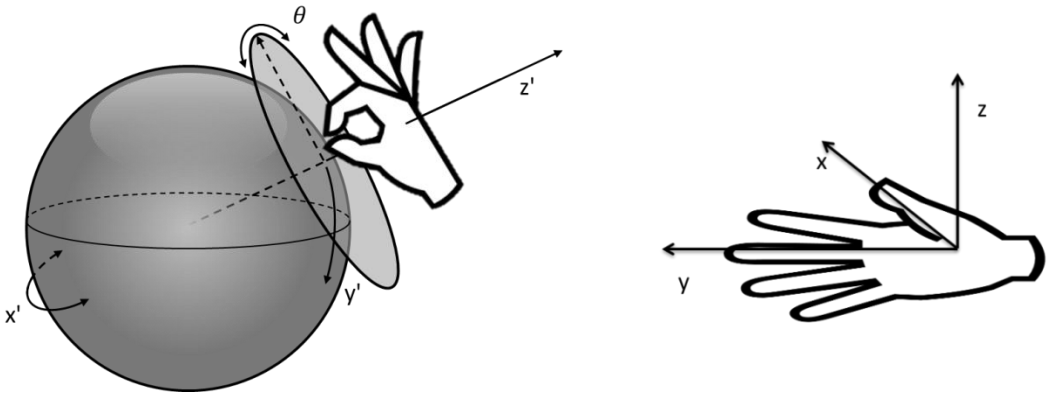

Figura 22. Liberdade de movimentos da mão. Imagem criada com base nos estudos publicados em (FERREIRA-BRITO e LANGEVIN, 1994).

3.2.4 Disposição das mãos (DM)

Na Libras, a disposição das mãos, no sentido de qual mão realiza qual gesto é importante. Como nota Ferreira-Brito (2010), certos gestos devem ser feitos utilizando-se a mão dominante, enquanto outros necessitam das duas mãos para serem realizados.

3.2.5 Orientação da palma das mãos (OM)

A orientação da palma das mãos é a direção a qual a palma das mãos está voltada durante a realização da articulação. Segundo Ferreira Brito (2010, p. 41) a orientação das palmas das mãos pode estar voltada para cima, para baixo, para esquerda, para direita, para frente ou para o corpo. A autora nota que também é possível ocorrer mudanças na orientação da palma durante a realização de um mesmo gesto.

$$P = \{\, cima, baixo, corpo, frente, esquerda, direita \,\}$$

3.2.6 Região de contato (RC)

A região de contato denota o modo com que a mão que gesticula o sinal entra em contato com o ponto de articulação, caso este contato ocorra. Segundo Ferreira-Brito (2010), exemplos de possíveis tipos de toque podem ser dados por um toque, um risco ou deslizamento. No entanto, a autora não especifica todas as possíveis regiões de contato. Nesta dissertação, consideraremos que o conjunto C seja finito, porém não conhecemos sua cardinalidade.

$$C = \{toque, duplo - toque, risco, deslizamento, \ldots\}$$

3.2.7 Componentes não manuais (NM)

Os componentes não manuais, como a expressão facial presente durante a execução do sinal desempenham papel determinante no significado final de cada gesto. Como evidenciam Silva *et al.* (2002), as expressões faciais em muitos casos

determinam o significado final do sinal, expressando a diferença entre uma afirmação, uma negação, um questionamento ou uma exclamação: Frases interrogativas geralmente envolvem um ligeiro movimento da cabeça para cima, enquanto que frases exclamativas envolvem um movimento para cima e para baixo. Frases negativas, em contrapartida, envolvem um característico movimento de um lado para o outro, em comum sinal de negação. Assim, as expressões faciais são comumente utilizadas para transformar as sentenças articuladas em perguntas, afirmações, pedidos, ou ordens (FERREIRA-BRITO, 2010).

Além de dar o significado a sentenças, a expressão facial também pode atuar como advérbio de modo ou intensificador numa frase da Libras. Como exemplificado por Strobel e Fernandes (1998), a mão aberta, com movimento lento e expressão serena denota *calma*. O mesmo sinal, porém com movimento brusco e expressão séria significa *pare*. Devido a sua alta variabilidade, é difícil dar uma forma específica ao conjunto de totais possíveis expressões corporais na Libras. No entanto, tendo como base os estudos de Baker-Shenk (1983), Ferreira-Brito (2010) conseguiu identificar 23 destas expressões, que utilizaremos para especificar o conjunto E da parametrização adotada conforme apresentado na Figura 23. Por fim, a autora também nota que algumas das expressões podem ser realizadas simultaneamente, como por exemplo, em uma indagação negativa (FERREIRA-BRITO, 2010, p. 242).

ROSTO
Sobrancelhas franzidas
Olhos arregalados
Lance de olhos
Sobrancelhas levantadas
Bochechas infladas
Bochechas contraídas
Lábios contraídos e projetados e sobrancelhas franzidas
Correr da língua contra a parte inferior interna da bochecha
Apenas a bochecha direita inflada
Contração do lábio superior
Franzir do nariz

CABEÇA
Balanço para frente e para trás (sim)
Balanço para os lados (não)
Inclinação para frente
Inclinação para o lado
Inclinação para trás

TRONCO
Para frente
Para trás
Balanço alternado de ombros
Balanço simultâneo de ombros
Balanço de um único ombro

ROSTO E CABEÇA
Cabeça projetada a frente, olhos levantados, cerrados (Que? Como? Quando? Por que?)
Cabeça projetada para trás, olhos arregalados (Quem?)

Figura 23. Conjunto de expressões faciais identificadas em (FERREIRA-BRITO, 2010, p. 240-241).

3.3 Gramática

A estrutura básica da gramática da Libras é a forma parametrizada do sinal apresentada na seção anterior. Além da formação de suas palavras, a gramática da Libras também pode ser estudada do ponto de vista de sua sintaxe e de suas classes gramaticais. Nesta seção, apresentaremos as principais características de sua sintaxe e outros elementos constituintes de sua gramática, como classes gramaticais e tipos de sentenças.

3.3.1 Sintaxe

Quanto à estrutura sintática de sua gramática, pode-se dizer que, na Libras, a ordem dos sinais na construção de uma frase obedece a regras refletindo a maneira de o surdo organizar suas ideias segundo sua percepção visual da realidade (STROBEL e FERNANDES, 1998). Segundo Quadros (2003), sentenças da forma Sujeito-Verbo-Objeto (SVO) são muito comuns na Libras e são sempre gramaticalmente aceitáveis. Já as sentenças nas formas Objeto-Sujeito-Verbo (OSV) e Sujeito-Objeto-Verbo (SOV) são permitidas quando há utilização de marcas não manuais, como as apresentadas na seção 3.2.7. As demais formas, segundo a autora, não existem na Libras. No entanto, apesar da forma SVO ser sempre gramaticalmente aceita, segundo Ferreira Brito (2010), a forma preferida na Libras seria a ordem tópico-comentário, ou topicalização, em que se coloca em destaque em primeiro lugar na frase o elemento sobre o qual se quer falar. A topicalização é muito frequente na fala coloquial do português oral, mas, na Libras, segundo a autora, quando não há restrições para seu uso, pode ser considerada regra geral.

3.3.2 Classes gramaticais

Assim como no Português e nas demais Línguas orais, na Libras também temos diversas classes gramaticais para as palavras articuladas. No entanto, segundo Nicoloso e Silva (2009), temos também a dificuldade de que, na Libras, pode ocorrer o caso de um mesmo sinal, articulado da mesma maneira, pertencer a classes gramaticais diferentes conforme o contexto da frase. Nesta seção, apresentaremos *algumas* das classes gramaticais da Libras.

3.3.2.1 Verbos

Os verbos em libras podem apresentar marca de concordância ou não. Verbos que apresentam marcação de concordância podem apresentar concordância número-pessoal, em que a orientação marca as pessoas do discurso; podem apresentar concordância de gênero, em que a configuração da mão marca o gênero; e podem apresentar concordância com a localização, em que o ponto de articulação marca a localização. Verbos que apresentam marcação de concordância são também chamados de verbos direcionais, pois a direção do movimento marca o sujeito e o objeto de acordo com seus pontos inicial e final, respectivamente (STROBEL e FERNANDES, 1998).

3.3.2.2 Adjetivos

Em Libras, os adjetivos possuem sempre forma neutra, e são, tipicamente, expressados no ar. A palavra "*redondo*", por exemplo, é expressa como um círculo desenhado no ar (Figura 24). Estes são exemplos de gestos icônicos, em que o gesto consiste na articulação da ação ou da forma do que se quer representar.

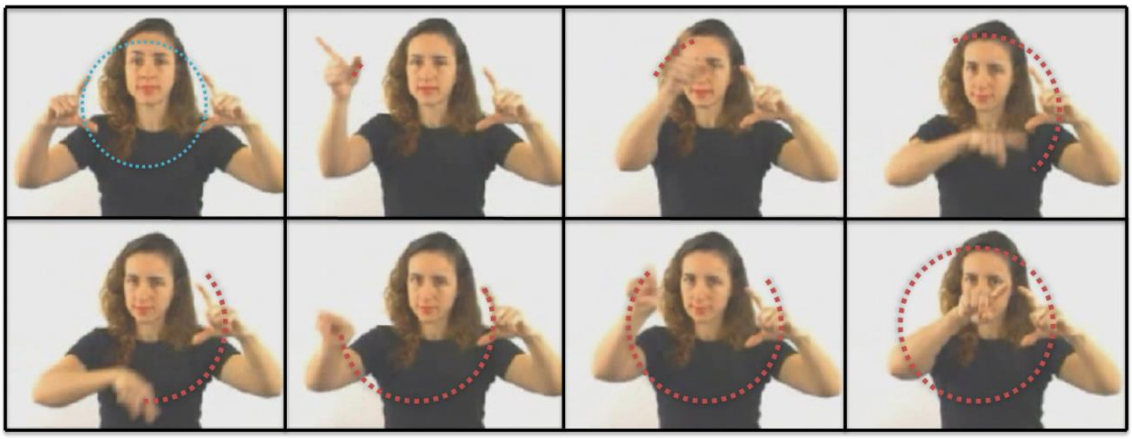

Figura 24. Expressão da palavra "redondo" em Libras. Adaptado de (LIRA e SOUZA, 2008).

3.3.2.3 Pronomes

Pronomes em Libras podem se apresentar nas formas *singular*, *dual*, *trial*, *quatrial* e *plural*. Na forma singular, a pessoa do discurso é apontada utilizando a configuração de mãos em ***d*** (indicador estendido). Na forma dual, as pessoas do discurso são apontadas utilizando-se a mão em forma de "2". Na forma *trial*, as pessoas são apontadas utilizando-se a forma de "3", e, na *quatrial*, formato de "4". Na forma plural, a sinalização pode ocorrer de duas maneiras: o sinal pode tanto ser composto pelo pronome em singular mais o sinal GRUPO; ou então o sinal para plural é feito pela mão dominante em configuração de ***d*** desenhando-se um círculo no ar.

3.4 Sentenças e frases

Na Libras, conforme mencionado na seção 3.2.7 (Componentes não manuais) as expressões faciais desempenham papel fundamental na especificação do tipo de frase que se quer comunicar. Exemplos são as frases afirmativas, negativas, exclamativas e interrogativas, que são marcadas por sinais específicos gesticulados com a face. Como levantado em (STROBEL e FERNANDES, 1998), podemos ter frases:

- *Afirmativas*, em que a expressão facial é neutra;
- *Negativas*, em que a negação pode ser feita utilizando-se um sinal de negativo específico, através de um movimento da cabeça de um lado para o outro em sinal de negação, através do acréscimo do sinal NÃO a frase afirmativa, ou a uma combinação das maneiras apresentadas;
- *Interrogativas*, em que a expressão facial apresenta sobrancelhas franzidas e há um ligeiro movimento da cabeça para cima;
- *Exclamativas*, em que a expressão facial apresenta sobrancelhas levantadas e há um ligeiro movimento da cabeça para cima e para baixo.

3.5 Alfabeto manual

A Libras, como outras Línguas de Sinais, também possui um alfabeto manual, utilizado em situações em que seja requerida a soletração de uma palavra, como um nome. O alfabeto manual também pode ser utilizado para representar cores através da sinalização da primeira letra da cor a que se quer referir. É possível notar diversas similaridades entre o alfabeto manual da Libras (Figura 25) e o alfabeto manual da ASL (Figura 26). No entanto, alfabetos manuais podem divergir significantemente entre si, como é o caso do alfabeto manual da BSL, o que evidencia ainda mais a não universalidade das Línguas de Sinais.

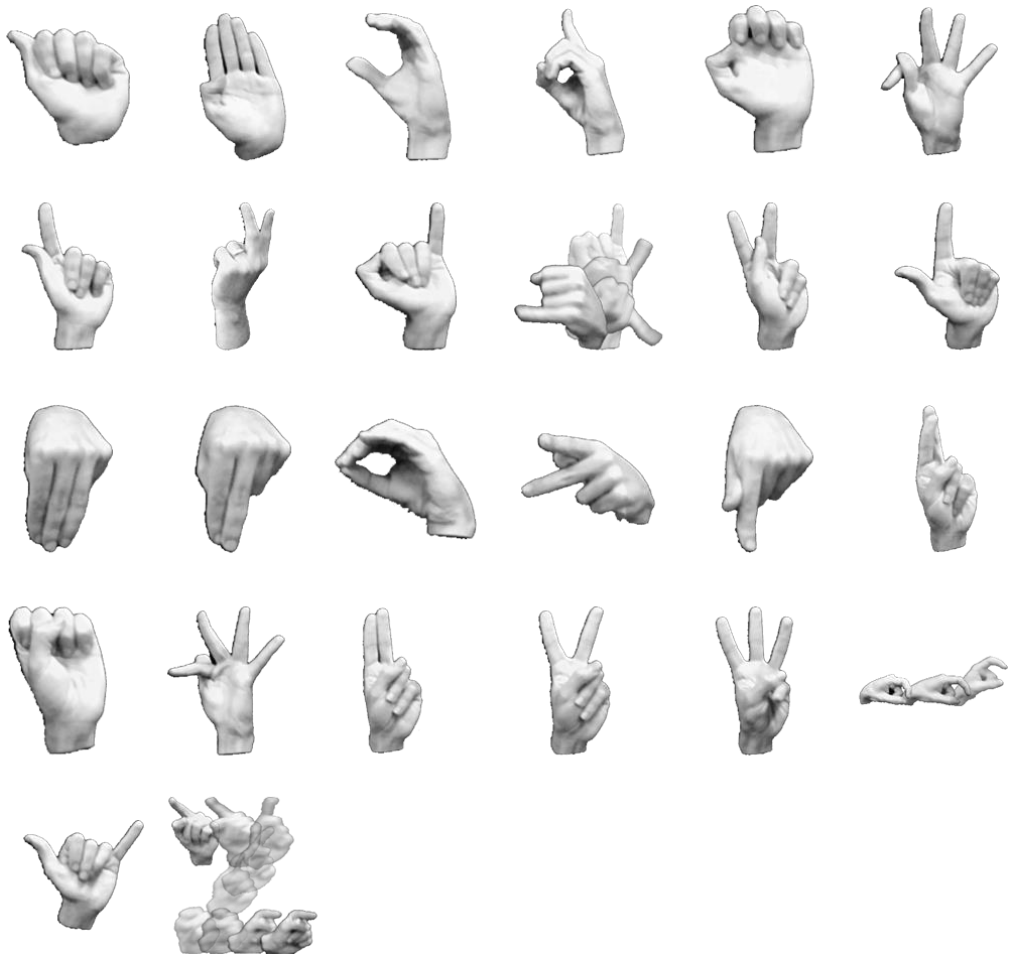

Figura 25. O alfabeto manual da Libras. Imagem criada a partir do banco de amostras coletado por Pedroso para uso em (PIZZOLATO, ANJO e PEDROSO, 2010).

Figura 26. Topo: o alfabeto manual da Língua de Sinais Britânica (BSL). Imagem criada utilizando-se a fonte tipográfica disponibilizada pela British Deaf Association. Baixo: alfabeto manual da Língua de Sinais Americana (ALS). Imagem criada utilizando-se a fonte tipográfica criada por David Rakowski, 1991.

3.6 Resumo do capítulo

Neste capítulo exploramos as principais características da Libras. Aqui, observamos a existência de severas dificuldades em seu reconhecimento, mesmo quando esta tarefa realizada por intérpretes humanos. Vimos como o sinal da Libras pode ser decomposto em unidades básicas, e verificamos que estas unidades formadoras estão potencialmente restritas a um conjunto finito de elementos. Em particular, verificamos que o espaço de sinalização está restrito a certas regiões ao redor do interlocutor, e que as configurações de mãos podem ser resumidas a um alfabeto relativamente pequeno de possibilidades.

No capítulo seguinte, exploraremos técnicas de visão computacional que potencialmente nos auxiliem a extrair informações sobre o que ocorre dentro do espaço de sinalização, explorando maneiras de detectar, segmentar e rastrear objetos em seu interior, como a configuração de mãos e movimentos do articulador.

Capítulo 4

Processamento de imagens e visão computacional

"All photos are accurate. None of them is the truth." — Richard Avedon

Diretamente relacionado ao problema de reconhecimento de gestos livres, no sentido de serem realizados sem o auxílio de luvas eletrônicas e outros dispositivos vestíveis, está o processamento de imagens e a visão computacional. A visão computacional une os campos de processamento de imagens e de reconhecimento de padrões. Apesar de muitos dos modelos oriundos desta última área serem diretamente aplicáveis ao problema de visão computacional, este capítulo se concentrará sobre problemas e soluções especializados em tratar problemas específicos da visão computacional.

Existem, na área de visão computacional, certos problemas recorrentes. Alguns dos principais problemas tratados pela área são os problemas de detecção de objetos, identificação de objetos e de rastreio de objetos em sequências de imagens. Neste capítulo detalharemos algumas das técnicas mais conhecidas para detecção, rastreio e identificação de objetos em imagens. Técnicas envolvendo aprendizado de máquina e reconhecimento de padrões a partir de aprendizado automático serão apresentadas e melhor detalhadas no capítulo seguinte, devidamente intitulado *Reconhecimento de Padrões*.

4.1 Representações de imagem

A visão computacional engloba métodos de aquisição, processamento, análise e entendimento de imagens. Neste contexto, o processamento de imagens digitais desempenha importante papel como base de muitos dos métodos envolvidos. A

escolha de uma representação adequada de uma imagem pode simplificar bastante seu processamento, como veremos na seção 4.3. A forma de aquisição da imagem também pode implicar mudanças significantes na natureza e quantidade de informação disponível em sua representação (Figura 27). Explorando informações alternativas ou complementares, pode-se obter maior sucesso na detecção de objetos específicos, como demonstra o uso de mapas de profundidade (Figura 28) para se realizar segmentação do corpo humano em tempo real como demonstrado pelo algoritmo de segmentação da plataforma de jogos *Xbox 360* para análise dos mapas de profundidade gerados pelo sensor *Kinect*.

O sensor *Kinect* é baseado na tecnologia de câmeras de profundidade desenvolvidas pela empresa israelense *PrimeSense*. Esta tecnologia interpreta informações de cenas 3D a partir de uma projeção contínua e estruturada de um padrão em luz infravermelha (FREEDMAN, SHPUNT, *et al.*, 2007). Este sistema de detecção de profundidade pode ser visto como uma variação da técnica de iluminação estruturada (SALVI, PAGÈS e BATLLE, 2004). O fato de o sensor *Kinect* utilizar emissores infravermelhos o torna imune a variações da luz ambiente.

A acurácia do sensor *Kinect* foi investigada por Khoshelham (2011), que também apresentou uma descrição detalhada de seu funcionamento. O sensor *Kinect* consiste em um emissor laser infravermelho, uma câmera infravermelha e uma câmera RGB. Seus inventores descrevem o processo de medição da profundidade como um processo de triangularização (FREEDMAN, SHPUNT, *et al.*, 2007) dos pontos emitidos. Seu emissor laser emite um único feixe que é subsequentemente dividido em múltiplos feixes seguindo uma grade de refração, criando um padrão luminoso projetado sobre o ambiente (Figura 29). Este padrão é então capturado pela câmera infravermelha e correlacionado contra um padrão de referência (KHOSHELHAM, 2011).

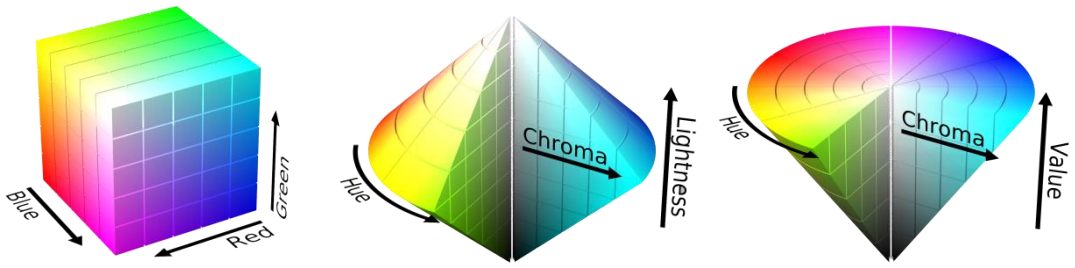

Figura 27. Diferentes espaços de cores renderizados sobre formas geométricas. Renderizações em POV-Ray por Michael Horvath, compartilhado sob licença Creative Commons.

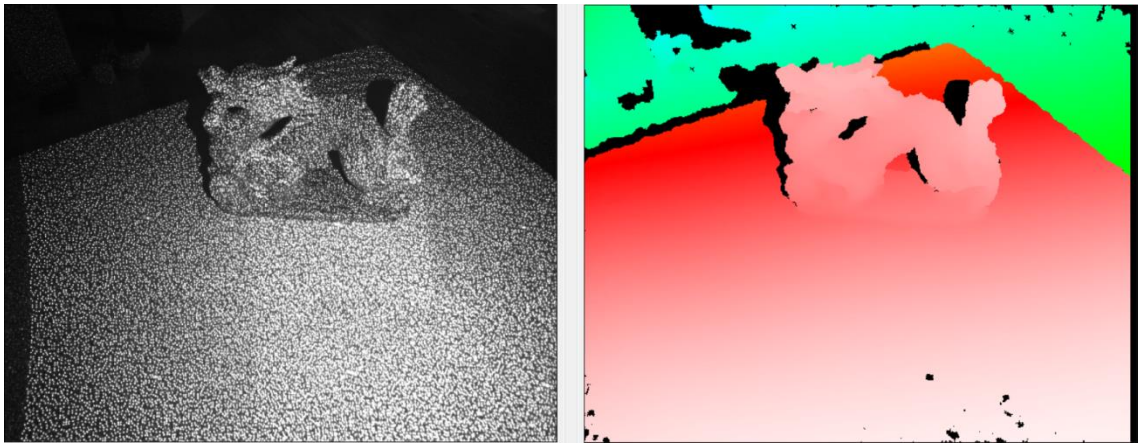

Figura 28. Saída do sensor Kinect. Esquerda: matriz de pontos projetados pelo emissor IR. Direita: Mapa de profundidade obtido através da análise da matriz de pontos projetada.

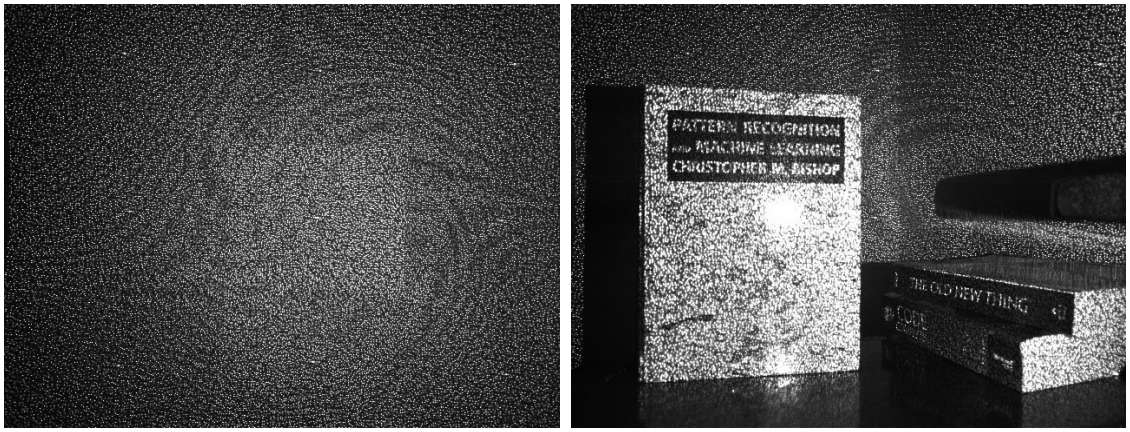

Figura 29. Padrão de pontos projetado pelo sensor Kinect.

Khoshelham também investigou a acurácia do sensor *Kinect* quando comparado a um sensor de varredura 3D profissional. Suas conclusões foram de que a nuvem de pontos de um sensor *Kinect* adequadamente calibrado não contem grandes erros sistemáticos, e que o erro aleatório das medidas de profundidade incrementa quadraticamente em relação ao aumento da distância do sensor. Este erro alcança 4 cm quando em profundidade máxima. A densidade dos pontos também decresce de acordo com o aumento da distancia do sensor. O autor também chega à conclusão de que a resolução do sensor *Kinect* é muito baixa quando mensurada em longa distância.

4.2 Segmentação de objetos

A segmentação é o primeiro passo em muitas aplicações de visão computacional. No reconhecimento de gestos, antes que seja possível aplicar técnicas de reconhecimento de padrões é primeiro necessário distinguir as regiões de interesse do restante da imagem, como as mãos e a face do restante do corpo ou do fundo da imagem, conforme demonstrado na Figura 30 a seguir.

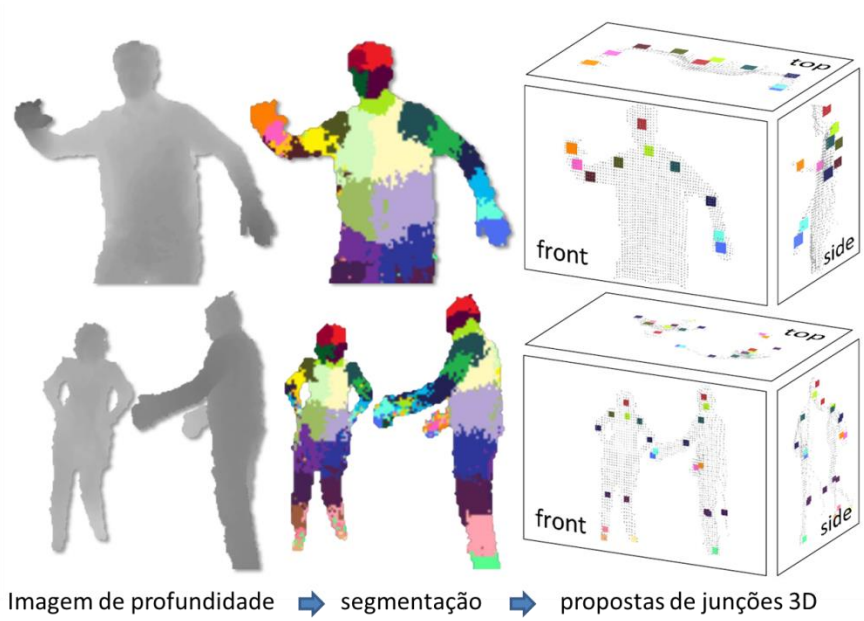

Figura 30. Distribuição de partes do corpo inferidas pelo método de (SHOTTON, FITZGIBBON, et al.*, 2011). Imagem adaptada de seu artigo original para incorporar traduções.*

O algoritmo de segmentação e detecção de objetos utilizado pelo console de jogos *Xbox 360* para interpretar as imagens coletadas pelo seu sensor *Kinect* foi apresentado em (SHOTTON, FITZGIBBON, *et al.*, 2011). Seu algoritmo de detecção é baseado em florestas de decisão aleatórias. Esta escolha de classificadores pode ser implementada de maneira eficiente em GPUs (SHARP, 2008), o que pode explicar a velocidade e baixo custo computacional desta abordagem. A abordagem utilizada não leva em conta os aspectos temporais da sequência de imagens, o que nos leva a concluir que o algoritmo não efetua rastreamento, mas sim a *classificação em tempo real de cada imagem capturada*. As características utilizadas no processo de decisão são extremamente simples, baseadas apenas em informações disponíveis

na imagem de profundidade. Dado um pixel $\boldsymbol{x}$ em uma imagem I, estas características são obtidas utilizando-se a expressão parametrizada

$$f_\theta(I, x) = d_I\left(x + \frac{\mathbf{u}}{d_I(x)}\right) - d_I\left(x + \frac{\mathbf{v}}{d_I(x)}\right) \tag{4.1}$$

em que $d_I(x)$ denota a profundidade do pixel $\boldsymbol{x}$ na imagem I e seus parâmetros $\boldsymbol{u}$ e $\boldsymbol{v}$ representam deslocamentos em relação a algum outro ponto da imagem. Como notam Shotton *et al.*, mesmo que estas características forneçam apenas um fraco sinal sobre qual parte do corpo o pixel se refere, o uso combinado da floresta de classificação torna possível a desambiguação de todas as partes sendo identificadas.

4.3 Detecção de objetos

Em 2001, Viola e Jones propuseram o primeiro framework de detecção de objetos em imagens capaz de operar em tempo real. Descrito em (VIOLA e JONES, 2001), seu trabalho foi demonstrado e parcialmente motivado pelo problema de detecção da face humana. Sua maneira de operação é baseada na busca exaustiva da imagem utilizando-se uma janela deslizante em diferentes escalas; algo que a primeira vista parece ser intratável, mas que possui uma elegante solução graças às contribuições introduzidas pelos autores.

Figura 31. Exemplo de detecção de faces utilizando o método de (VIOLA e JONES, 2001).

As contribuições oriundas do trabalho de Viola-Jones se resumem basicamente a três inovações. A primeira foi a representação da imagem no formato de *Imagem Integral*. Note que o conceito de *Imagem Integral* não era necessariamente novo, sendo conhecido sob o nome de Tabela de Áreas Somadas (*Summed Area Table*, SAT). No entanto, sua aplicação era tradicionalmente relacionada ao mapeamento de texturas no campo de computação gráfica (CROW, 1984). Sua utilização como base de representação das entradas de um classificador possibilitou a computação de características de maneira extremamente eficiente.

A segunda contribuição do trabalho destes autores foi a modificação do algoritmo de *AdaBoost* para gerar classificadores extremamente simples, baseados em uma única característica. Assim, a modificação do algoritmo *AdaBoost* de aprendizado possibilita que o método passe a ser visto também como um processo de seleção de atributos.

A terceira e igualmente importante contribuição dos autores foi a inovação em restringir os classificadores numa estrutura em cascata, de forma que regiões não interessantes da imagem pudessem ser descartadas rapidamente, sem desperdiçar processamento (Figura 32). Desta maneira, os primeiros classificadores na cascata podem ser vistos como filtros de atenção, possibilitando que a parte mais pesada do processamento se concentre apenas sobre regiões que potencialmente contenham uma face ou outro objeto em questão.

Mesmo utilizando classificadores extremamente simples, baseados em uma única característica cada um, o método é capaz de apresentar desempenho surpreendente e pouquíssimos casos de falsos positivos. As características utilizadas pelos classificadores são as chamadas características de *Haar* (Figura 33), que tem sua origem nas funções bases de *Haar* utilizadas por Papageorgiou, Oren e Poggio (1998). No entanto, apesar de toda teoria justificando sua origem, estas características podem ser explicadas simplesmente como diferenças de intensidade entre diferentes áreas da imagem. Em termos gerais, os classificadores buscam apenas regiões na imagem cuja diferença de intensidade entre si seja maior do que algum limiar de ativação, preferencialmente aprendido de maneira automática.

Certa intuição por detrás da motivação pela busca de regiões de diferentes intensidades pode ser explicada pela Figura 34 a seguir. O classificador de Viola-Jones percorre a imagem exaustivamente através de uma janela deslizante, considerando múltiplas transformações de escala desta janela. Como é possível observar no caso específico da detecção de faces, características de dois e três retângulos podem prontamente casar com regiões específicas do rosto humano.

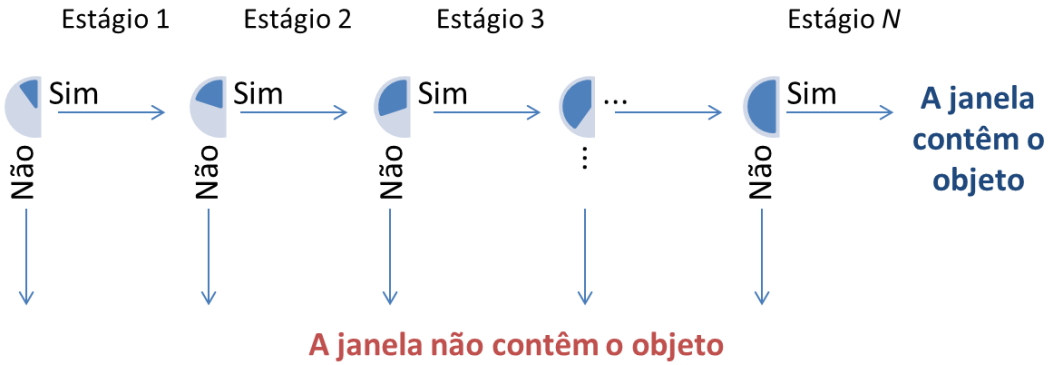

Figura 32. Diagrama representativo da cascata de classificadores fracos proposto por (VIOLA e JONES, 2001).

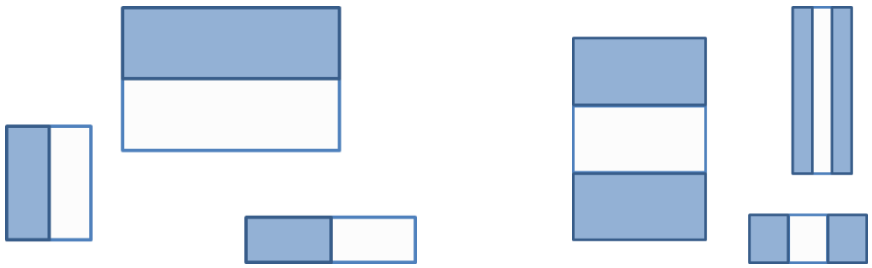

Figura 33. Esquerda: diferentes características de Haar de dois retângulos. Direita: diferentes características de Haar de três retângulos.

Figura 34. Detector de objetos deslizando sobre a imagem em diferentes escalas. Imagem central e direita mostram potenciais regiões de casamento para as características de Haar descrevendo o rosto humano.

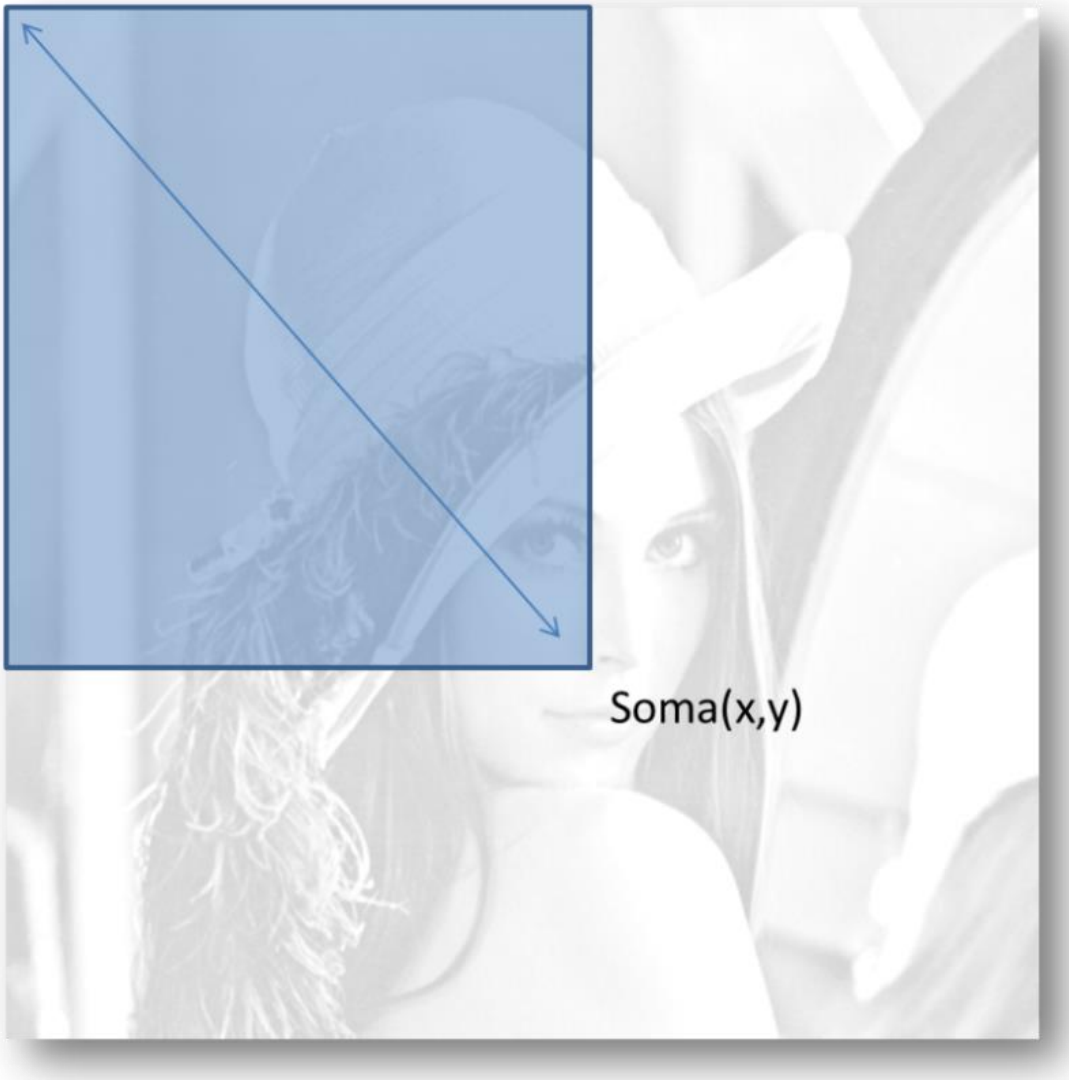

Figura 35. Representação em imagem integral, em que cada elemento desta representação equivale à soma de todos os pixels na imagem original desde o canto superior esquerdo até a posição do elemento corrente.

Uma observação muito interessante sobre o método vem do ponto de vista biológico. As características retangulares de *Haar* parecem identificar características similares às identificadas pelas células neuronais de campos receptivos *complexos* da área 17 do córtex visual felino (HUBEL e WIESEL, 1962)[9].

Outra característica interessante do método de Viola-Jones é o fato de que cada estágio de classificação opera apenas marginalmente melhor do que um classificador aleatório. Cada classificador em seu framework de detecção é considerado um classificador fraco. Porém, uma vez combinados através da técnica de *Boosting*, estes se tornam um classificador forte capaz de apresentar altas taxas de sucesso em classificações. O classificador final proposto no trabalho original de Viola-Jones consistia em uma cascata de 32 camadas, incluindo um total de 4.297 características. O primeiro estágio da cascata era construído utilizando-se duas características, conseguindo rejeitar corretamente 60% de regiões não contendo faces e detectar corretamente 100% das regiões contendo faces.

No entanto, considerando que o classificador utiliza uma janela deslizante para testar cada região da imagem, ainda que as características sejam baseadas apenas na simples diferença de intensidade entre duas regiões da imagem seria necessário significante esforço para recalcular estas áreas de intensidade para todas as diferentes posições da janela deslizante. Neste ponto voltamos à primeira contribuição do trabalho de Viola e Jones, a representação em imagem integral. Antes de iniciar o deslize da janela pela imagem, a imagem original é transformada para a representação em imagem integral, representada na Figura 35. Uma vez convertida, a soma dos pixels em uma janela retangular arbitrária pode ser computada facilmente utilizando apenas quatro acessos à memoria, um ganho considerável em comparação a recalcular sua área percorrendo todos os pixels desta janela.

A representação em imagem integral pode ser descrita por

$$\mathrm{ii}(x, y) = \sum_{u=1}^{x} \sum_{v=1}^{y} \mathrm{i}(u, v) \tag{4.2}$$

[9] O córtex visual é constituído de uma variedade de tipos celulares, cada tipo sendo especializado em analisar diferentes características visuais. Células do tipo complexo são especializadas em detectar bordas orientadas em ângulos específicos, como barras horizontais ou verticais. Estas células são receptoras diretas das chamadas células simples, que simplesmente reagem a padrões circulares de luminosidade como as células da retina.

em que i é a imagem original e ii representa a imagem integral. Apesar de parecer bastante custosa, esta computação pode ser realizada de maneira eficiente em apenas um único passo sobre a imagem utilizando o par de recorrências

$$\begin{aligned} s(x,y) &= s(x,y-1) + i(x,y) \\ ii(x,y) &= ii(x-1,y) + s(x,y) \end{aligned} \tag{4.3}$$

ou, simplesmente,

$$ii(x,y) = i(x,y) + ii(x-1,y) + ii(x,y-1) - ii(x-1,y-1) \tag{4.4}$$

com a definição adicional de que $s(x,-1) = ii(-1,y) = 0$. Utilizando a imagem integral, qualquer soma retangular na imagem pode ser computada apenas em quatro referências à memória. A soma de um retângulo indo de (x_a, y_a) até (x_b, y_b) pode ser calculada por

$$\sum_{x=x_a}^{x_b} \sum_{y=y_a}^{y_b} i(x,y) = ii(x_b,y_b) - ii(x_{a-1},y_b) - ii(x_b,y_{a-1}) + ii(x_{a-1},y_{a-1}) \tag{4.5}$$

conforme evidenciado na Figura 36.

Desta maneira, cada característica de *Haar* requer apenas um número constante de acessos a memoria de acordo com sua definição. Uma característica de dois retângulos, por exemplo, necessitaria de seis acessos à memória para ser calculada. Uma característica de três retângulos, em contrapartida, necessitaria oito. É interessante notar que a representação em imagem integral também é utilizada pelos algoritmos de detecção de pontos de interesse SURF (BAY, ESS, *et al.*, 2008; EVANS, 2009) e FREAK (ALAHI, ORTIZ e VANDERGHEYNST, 2012).

Após a publicação de Viola e Jones, Lienhart e Maydt (2002) propuseram uma extensão do método original, introduzindo novas características, como as características de *Haar* inclinadas capazes de indicar a ocorrência de bordas orientadas a 45°, e tornando cada estágio de classificação uma árvore de decisão ao invés de um simples nó. Tendo em vista que o método de Viola-Jones é patenteado, esta leve modificação, além de tornar o classificador mais poderoso, o retira da cobertura da patente original, evitando que para seu uso comercial seja necessário adquirir uma licença dos detentores da patente.

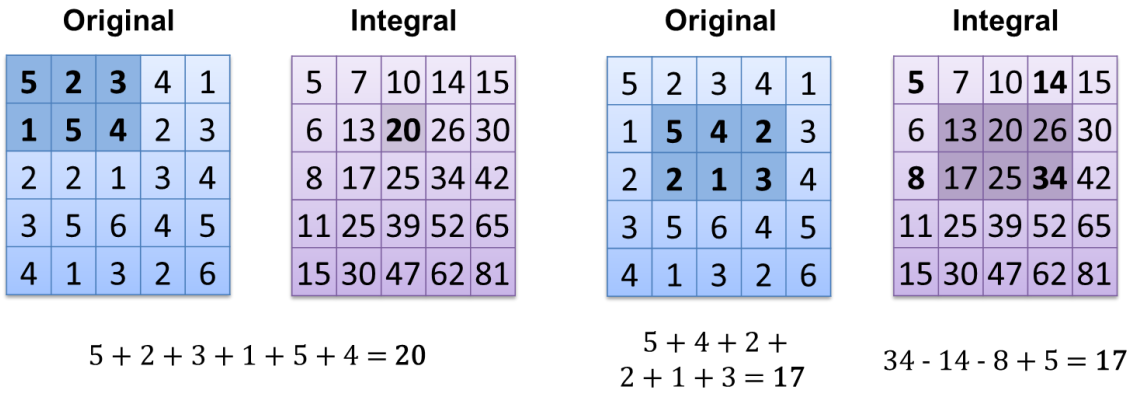

Figura 36. Esquerda: Exemplo de cálculo da representação em imagem integral. Direita: Cálculo de uma área retangular utilizando a imagem integral.

Como comentário final nesta seção, é interessante relembrar que, como mencionado ao final da seção 2.3, uma das outras abordagens de maior sucesso no reconhecimento de expressões faciais é baseada no uso de características extraídas de filtros de Gabor e classificadas utilizando-se SVMs (BARTLETT, LITTLEWORT, *et al.*, 2003). No entanto, Whitehill e Omlin (2006) mostraram que, apesar do sucesso desta abordagem, seu custo computacional em termos de memória e tempo de processamento é bastante alto. Em contrapartida, sua investigação mostrou que o uso de características de *Haar* classificadas por um algoritmo *AdaBoost* é capaz de alcançar taxas de sucesso similares, porém operando mais rapidamente por várias ordens de magnitude.

4.4 Rastreio de objetos por distribuição de cores

O algoritmo *Camshift*, apresentado em (BRADSKI, 1998), é um algoritmo de rastreio baseado em cores. O uso de simples distribuições de cores como atributo principal o torna um algoritmo bastante rápido e simples de ser implementado. Testes envolvendo o rastreio de faces revelam que o algoritmo também é robusto e capaz de se recuperar mesmo na presença de oclusões. Nesta seção, apresentaremos o algoritmo *Camshift* com maiores detalhes.

O nome *Camshift* vem de *Continuously Adaptive Mean Shift*, ou deslocamento adaptativo e contínuo da média. Este algoritmo é uma extensão do algoritmo de deslocamento da média (*Mean Shift*) para estimação da moda em distribuições

de probabilidade[10]. O algoritmo de deslocamento de média emprega uma abordagem não paramétrica para alcançar seu objetivo através de uma subida de gradiente. O algoritmo é robusto no sentido que valores discrepantes, não pertencentes à distribuição, mas ainda assim presentes no conjunto de dados, não surtem grandes efeitos no resultado final do algoritmo.

Como o algoritmo de deslocamento de média opera sobre distribuições de probabilidade, torna-se necessário modelar as cores presentes num quadro da sequência de imagens como probabilidades. Para isto, são utilizados histogramas de cores e a técnica de retroprojeção de histograma. O primeiro histograma de cores é armazenado durante a fase de inicialização do algoritmo. Sendo um método puramente de rastreio, o algoritmo deve ser inicializado com uma região inicial que deve ser rastreada na imagem, conforme apresentado na Figura 37.

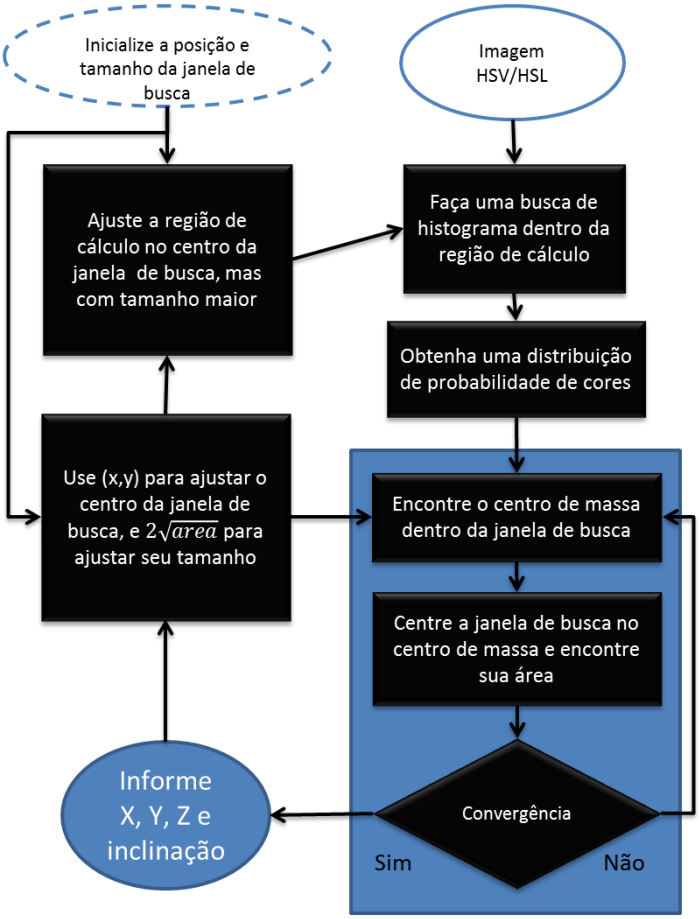

Figura 37. Algoritmo Camshift, conforme proposto por Bradski em sua publicação original (1998).

[10] A moda de uma distribuição é definida como seu valor com maior probabilidade de ocorrência. No caso de distribuições unimodais, como a distribuição Gaussiana, a média, a mediana e a moda coincidem todas neste mesmo valor de pico.

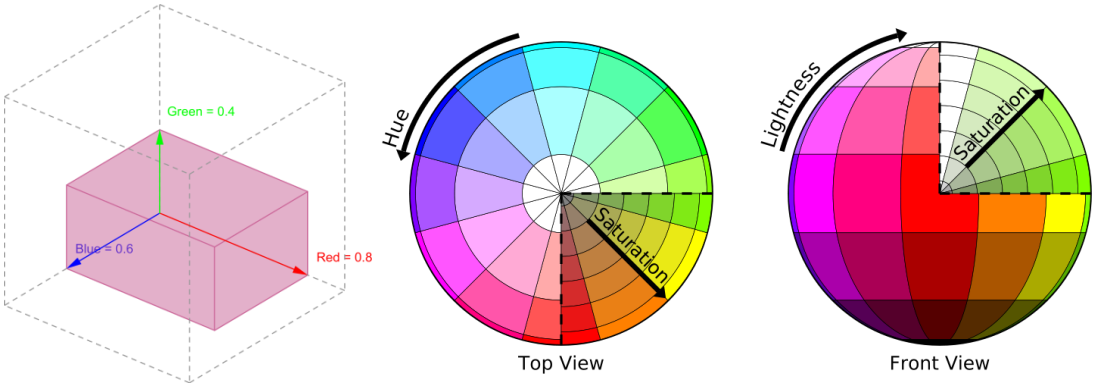

Figura 38. Diferentes espaços de cores. Esquerda: espaço de cores RGB. Direita: espaço de cores HSL mapeado para uma esfera, com corte de canto. Criada por Michael Horvath e compartilhada sob a licença Creative Commons.

Assim que uma região de interesse é passada para o algoritmo, este se inicia computando um histograma de cores para esta região. Um histograma de cores nada mais é do que um vetor contendo a contagem de cada possível valor de cores ao longo de uma imagem. O valor de cor específico depende do espaço de cores em que a imagem está sendo representada. Exemplos de diferentes espaços de cores são os espaços RGB, HSL e HSV, exibidos na Figura 38.

Em sua publicação, Bradski considerou apenas o componente Matiz (*Hue*) do espaço de cores HSV para montar seu histograma. O uso da matiz é convidativo, pois neste espaço de cores os componentes Matiz, Saturação e Brilho estão descorrelacionados, possibilitando que a distribuição de cores apresente certa independência à luminosidade ambiente ou a diferentes tons de uma mesma matiz (como é o caso dos diferentes tons de pele). É interessante notar que há um problema inerente ao uso exclusivo do componente matiz no cálculo dos histogramas de cores. Para tons de luminosidade muito altos ou muito baixos, não há matiz definida, já que qualquer valor de matiz poderia produzir a ausência ou presença completa de luz. Desta maneira, é necessário cortar ou ignorar valores cujos componentes de saturação e brilho estejam em certos intervalos.

Uma vez que o histograma tenha sido computado no estágio de inicialização do algoritmo, quadros subsequentes tem o valor de seus pixels transformados em probabilidades através do cálculo da razão de histogramas: o histograma do quadro atual é computado, a razão de cada elemento em relação ao histograma inicial é interpretada como uma probabilidade, e o valor de cada pixel no quadro é então substituído por sua respectiva probabilidade. A esta operação dá-se o nome de *retroprojeção de histograma*, cujo resultado para uma imagem RGB utilizando-se um rosto humano como objeto inicial é apresentado na Figura 39.

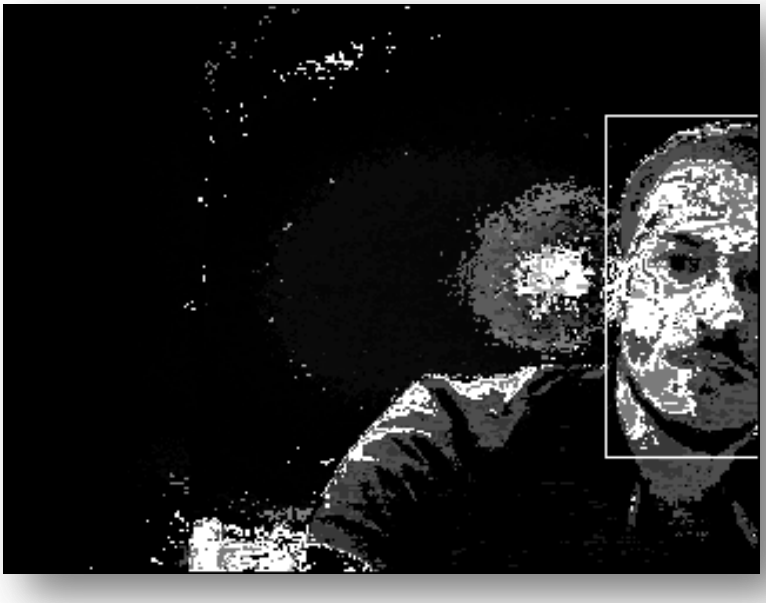

Figura 39. Retroprojeção de histograma em imagem RGB

Assim que a retroprojeção do histograma tenha sido computada e tenhamos obtido o mapa de probabilidades para a imagem, podemos executar o algoritmo *Camshift* para deslocar a estimativa inicial de posição do objeto para seu novo centro de massa. O centro de massa é computado através do método iterativo de deslocamento de média (FUKUNAGA e HOSTETLER, 1975). O método trabalha a partir da computação dos primeiros e segundos momentos da imagem, e desloca a estimativa atual da média até que este deslocamento seja menor do que algum limite pré-definido ou algum número máximo de iterações. Este passo pode ser visto como a estimação de uma Gaussiana bidimensional sobre o mapa de probabilidades 2-D. É compreensível esperar que este método funcione adequadamente ao problema de rastreio de faces dado a inerente regularidade no formato elíptico e, portanto, aproximadamente Gaussiano da face humana.

Para a estimação do centroide desta distribuição 2-D, o deslocamento de média utiliza os primeiros momentos da imagem. O momento zero, o primeiro momento para a coordenada x, e o primeiro momento para a coordenada y do mapa de probabilidades $I(x, y)$ são definidos respectivamente como

$$M_{00} = \sum_x \sum_y I(x,y) \quad M_{10} = \sum_x \sum_y I(x,y)x \quad M_{01} = \sum_x \sum_y I(x,y)y \tag{4.6}$$

em que $I(x, y)$ se limita a valores dentro da janela de busca atual. O novo centroide a ser utilizado como localização do centro da janela de busca do algoritmo será então dado por

$$x_c = \frac{M_{10}}{M_{00}} \qquad y_c = \frac{M_{01}}{M_{00}}. \tag{4.7}$$

Após estes valores serem encontrados, a janela se desloca para estas novas posições. O fator limitante do algoritmo de deslocamento de média que o inviabiliza para distribuições dinâmicas, ou seja, em movimento, como objetos em movimento em uma cena, é o fato de que o algoritmo não atualiza o tamanho de sua janela de busca, apenas sua posição. O algoritmo *Camshift* resolve este problema utilizando o momento zero obtido durante a avaliação do deslocamento de média para guiar a atualização do tamanho da nova janela. O momento zero pode ser interpretado como a área do objeto identificada na região da janela. Assim, o raio ou a altura e largura da janela podem ser estimados em função da área e de outras restrições, como alguma razão de proporção pré-definida.

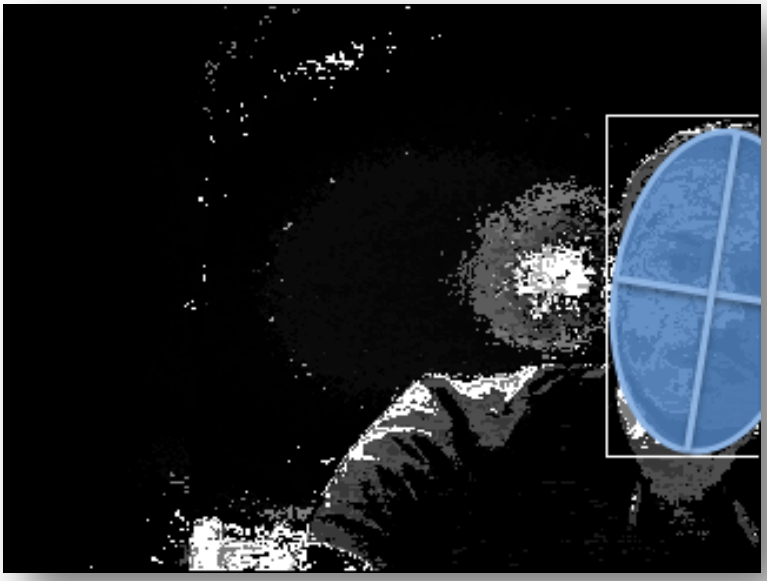

Figura 40. Exemplificação aproximada da Gaussiana bidimensional estimada sobre um objeto de rastreio; no caso, uma face humana.

Através dos segundos momentos da imagem, é possível obter uma estimativa da inclinação da elipse contendo a distribuição de probabilidade do objeto. Os segundos momentos da imagem são definidos para a região atual da janela do mapa de intensidades $I(x, y)$ como:

$$\begin{aligned} M_{20} &= \sum_x \sum_y I(x,y)x^2 \quad \text{Segundo momento para coordenada x} \\ M_{02} &= \sum_x \sum_y I(x,y)y^2 \quad \text{Segundo momento para coordenada y} \end{aligned} \tag{4.8}$$

Assumindo que o objeto possua uma forma elíptica, é razoável supor que sua distribuição de probabilidade bidimensional no mapa de probabilidades possa assumir uma forma Gaussiana. Deste ponto de vista, podemos obter a forma desta distribuição a partir de sua matriz de covariância. A matriz de covariância para esta distribuição será dada por

$$\Sigma = \begin{pmatrix} a & b/2 \\ b/2 & c \end{pmatrix} \tag{4.9}$$

em que

$$a = \frac{M_{20}}{M_{00}} - x_c{}^2 \quad b = \frac{M_{11}}{M_{00}} - x_c y_c \quad c = \frac{M_{02}}{M_{00}} - y_c{}^2 \tag{4.10}$$

O comprimento e a largura do maior eixo principal podem então ser computados a partir dos autovalores de Σ. Para o caso bidimensional, o comprimento l e a largura w podem ser definidos por

$$\begin{aligned} l &= \sqrt{\frac{(a+c) + \sqrt{b^2 + (a-c)^2}}{2}} \\ w &= \sqrt{\frac{(a+c) - \sqrt{b^2 + (a-c)^2}}{2}} \end{aligned} \tag{4.11}$$

e a inclinação desta elipse pode ser facilmente encontrada considerando o ângulo de orientação do auto vetor associado com o maior autovalor, dado por

$$\theta = \frac{1}{2}\arctan\left(\frac{2b}{a-c}\right). \tag{4.12}$$

Assim, podemos verificar que é possível empregar o algoritmo *Camshift* para rastrear quatro graus de liberdade em uma sequência de imagens. Quando aplicado a faces, estes graus podem ser definidos como a posição x, y da cabeça do usuário na cena, sua proximidade a câmera z, estimada a partir da área da imagem de probabilidade, e a inclinação de seu rosto θ, ilustrados na Figura 41 a seguir.

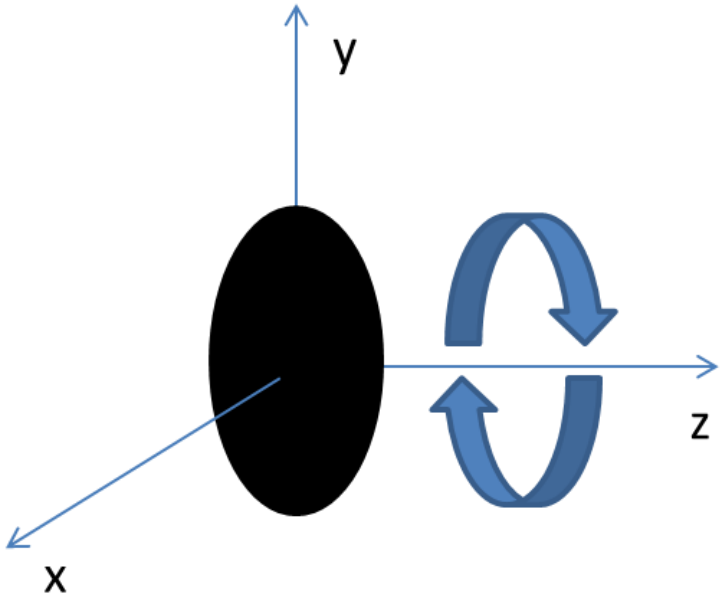

Figura 41. Graus de liberdade rastreados pelo algoritmo Camshift

Finalizando esta seção, podemos notar que o uso de um algoritmo baseado em cores é motivado por sua simplicidade computacional. A formulação simples do algoritmo *Camshift* evita o uso de métodos de rastreio mais elaborados como filtros de *Kalman* ou modelos preditivos. A habilidade do *Camshift* em ignorar valores discrepantes o permite ser razoável para aplicações de rastreio de face, a um custo computacional baixo, o que permite deixar maior processamento para outras tarefas. Na próxima seção, veremos uma possibilidade de combinar rastreadores baseados em casamento de modelos para expandir o número de graus de liberdade possíveis de serem rastreados em uma imagem.

4.5 Rastreio de objetos por casamento de modelos

A detecção de objetos por casamento de modelos (ou *templates*) é um dos métodos mais comuns e extremamente simples de se encontrar um objeto similar em uma imagem. Na abordagem de casamento de modelos, o objeto a ser identificado na imagem é separado e denominado modelo ou *template*. A seguir, conduz-se uma busca exaustiva (Figura 42) na imagem de maneira a encontrar uma sub-região da imagem em que a distância entre o conteúdo desta sub-região e o modelo seja mínima. A busca pelo modelo é facilmente paralelizável tendo em vista que cada pixel é analisado separadamente.

Figura 42. Casamento de modelo (template) por busca exaustiva.

Tipicamente, possíveis escolhas para esta distância incluem a soma de diferenças quadráticas (*Sum of Squared Differences*, SSD), a soma de diferenças absolutas (*Sum of Absolute Differences*, SAD) ou ainda a soma de diferenças transformadas absolutas (*Sum of Absolute Transformed Differences*, SATD). Estas distâncias podem ser calculadas pelas fórmulas

$$\begin{aligned} d_{SSD}(u,v) &= \sum_x \sum_y (f(x,y) - t(x-u, y-v))^2 \\ d_{SAD}(u,v) &= \sum_x \sum_y |(f(x,y) - t(x-u, y-v))| \\ d_{SATD}(u,v) &= \sum_x \sum_y \left| H * \left(f(x,y) - t(x-u, y-v)\right) * H^T \right| \end{aligned} \tag{4.13}$$

em que f denota a imagem a ser percorrida e t denota o modelo utilizado. No caso SATD, H denota uma matriz de transformação, sendo a matriz de *Hadamard* a opção mais comum. Quanto menor o valor de cada uma das métricas, mais similares são a região atual e o modelo.

Podemos ver que a SSD é simplesmente a distância Euclidiana quadrática entre a sub-região da imagem e o modelo. Também podemos notar que a mais simples de todas as distâncias é a SAD, que pode ser vista como o simples cálculo da norma-L1 da imagem de diferenças. Devido a sua simplicidade, a métrica SAD é extremamente rápida de ser computada. Além do mais, certos processadores possuem instruções específicas para se calcular este tipo de métrica, como a instrução PSADBW (*Packed Sum of Absolute Differences*) pertencente ao conjunto de instruções Intel (INTEL CORPORATION, 2003) e AMD (ADVANCED MICRO DEVICES, INC., 2000).

Outra maneira de se formular o problema é considerar uma medida de correlação cruzada. Expandindo a distância SSD, obtemos a expressão

$$\begin{aligned} d_{SSD}(u,v) &= \sum_x \sum_y (f(x,y) - t(x-u, y-v))^2 \\ &= \sum_x \sum_y (f^2(x,y) - 2f(x,y)t(x-u, y-v) + t^2(x-u, y-v)) \end{aligned} \tag{4.14}$$

na qual pode-se observar que o termo $\sum_x \sum_y t^2(x-u, y-u)$ é sempre constante. Assumindo que $\sum_x \sum_y f^2(x,y)$ também seja constante, é possível obter a expressão da medida de correlação cruzada para casamento de modelos

$$c(u,v) = \sum_x \sum_y f(x,y)t(x-u, y-v) \tag{4.15}$$

em que $c(u,v)$ agora representa uma medida de similaridade, ao invés de uma distância como as demais medidas. Existem, no entanto, muitas desvantagens em se utilizar a correlação cruzada como apresentada acima para o casamento de modelos. Uma destas dificuldades se origina da hipótese de que a energia das regiões da imagem $\sum_x \sum_y f^2(x,y)$ seja constante, o que não é necessariamente sempre verdade. Esta e outras dificuldades, como as levantadas por Lewis (1995) levaram a especificação da correlação cruzada normalizada

$$\gamma(u,v) = \frac{\sum_x \sum_y (f(x,y) - \bar{f}_{u,v})(t(x-u, y-v) - \bar{t})}{\sqrt{\sum_x \sum_y (f(x,y) - \bar{f}_{u,v})^2 \sum_x \sum_y (t(x-u, y-v) - \bar{t})^2}} \tag{4.16}$$

em que $\bar{t}$ é a média do modelo e $\bar{f}_{u,v}$ é a media da região sob o modelo. O uso da correlação cruzada é particularmente atraente, pois sua formulação pode ser vista como uma convolução no domínio do espaço entre a imagem f e o modelo reverso $t'(x,y) \equiv t(-x,-y)$ como

$$(f * t')(u,v) = \iint_{-\infty}^{+\infty} f^*(x,y)t'(x-u, y-v)\, dxdy, \tag{4.17}$$

o que torna a correlação cruzada passível de ser realizada de maneira eficiente no domínio da frequência através da propriedade da multiplicação no domínio da frequência/convolução no domínio do espaço, sumarizada nas equações

$$\begin{aligned} \mathcal{F}\{f * t\} &= \mathcal{F}\{f\} \cdot F\{t\} \\ f * t &= \mathcal{F}^{-1}\{\mathcal{F}\{f\} \cdot F\{t\}\}. \end{aligned} \tag{4.18}$$

No entanto, apesar desta propriedade ser bem conhecida, a forma normalizada da correlação cruzada, que é preferida nas aplicações de casamento de modelo

não tem uma expressão simples no domínio da frequência. O trabalho de Lewis (1995) se dedicou a mostrar como a correlação cruzada não normalizada pode ser normalizada de maneira eficiente utilizando a representação em Imagem Integral (apresentada na seção 4.3) sobre a janela de busca[11]. A transformação ao domínio da frequência pode ser computada de maneira eficiente utilizando-se o algoritmo da Transformada Rápida de Fourier (*Fast Fourier Transform*, FFT), originalmente apresentado em (COOLEY e TUKEY, 1965) e posteriormente continuamente popularizado e melhorado nas décadas seguintes (HASSANIEH, INDYK, *et al.*, 2012).

Dada à característica do casamento de modelo ter uma distância (ou uma medida de similaridade) de fácil acesso trás inspiração para a elaboração de um rastreador baseado em casamentos sucessivos de *template*. Um método dinâmico de rastreamento baseado casamento de modelos pode ser visto como um algoritmo de casamento de modelos que seja capaz de atualizar o modelo ou a região de interesse da busca de acordo com informações de quadros anteriores. A utilização de métodos como a SAD possibilita que seja identificado, juntamente com o modelo, o valor de similaridade entre o modelo original e a melhor região identificada na imagem. Esta possibilidade dá margem para a criação de um algoritmo com reinicialização automática que seja capaz de retomar o processamento sempre que a imagem de rastreio for perdida. Um algoritmo desta natureza pode ser uma alternativa a métodos baseados na distribuição de cores como o *Camshift*, que teriam problemas rastreando um objeto de cor muito similar a seu fundo. Esta abordagem é extremamente simples por evitar o uso de mecanismos custosos como filtros de *Kalman*, modelos 3D ou outras abordagens probabilísticas.

Algoritmos de rastreio baseados em casamento de modelos não são aplicáveis a qualquer problema, pois o casamento tende a ser muito custoso. No entanto, se a busca puder ser limitada apenas a uma área relativamente pequena, o que é caso quando consideramos a janela obtida pelo rastreamento anterior, seu tempo de computação torna-se aplicável a tempo real. A maior dificuldade nesta abordagem ocorre quando há uma mudança muito brusca na janela de busca, o que faz com que o rastreador se perca. Neste ponto, ressalta-se a importância da detecção da perda do objeto de rastreio, para que o algoritmo de detecção assuma o controle e reencontre a região de interesse para o rastreador.

[11] Tal técnica foi utilizada por J. P. Lewis, enquanto trabalhava para a Industrial Light & Magic, na produção de *Forrest Gump (1994)*.

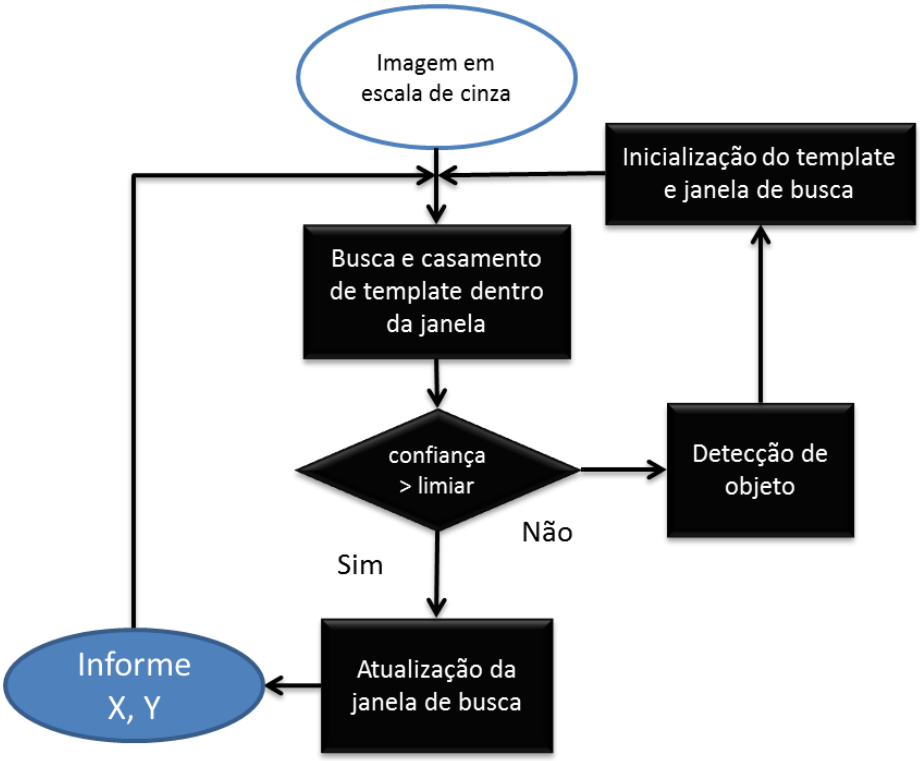

Figura 43. Fluxograma descrevendo o rastreamento de objetos através do casamento de templates.

Algoritmos de rastreio baseados em casamento de modelos aplicáveis a tempo real foram propostos em trabalhos como a publicação de Jurie e Dhome (2002). Em seu trabalho, os autores propõem um algoritmo de rastreio baseado em casamento de modelos capaz de rastrear alvos em tempo real, devotando especial atenção a oclusões e variações de iluminação. Apresentam um algoritmo para tratar oclusões e variâncias de iluminação em rastreadores baseados na SSD. Sua abordagem consiste em representar o modelo como uma pirâmide de submodelos, rastreando independentemente cada padrão. Esta estimativa se torna robusta por ser baseada em movimentos locais ao invés de apenas intensidades, o que leva a uma supressão de possíveis ambiguidades. Como afirmam os autores, esta abordagem em submodelos provém um balanço entre robustez e acurácia, em que submodelos menores proveem uma trajetória inacurada, porém robusta, e os maiores proveem uma trajetória não robusta, porém acurada.

4.6 Resumo do capítulo

Este capítulo apresentou e descreveu técnicas de processamento de imagens e visão computacional que podem ser utilizadas para localizar e rastrear humanos e objetos em imagens de vídeo. Aqui, vimos como as técnicas de detecção de objetos baseadas em características de Haar são capazes de realizar a detecção de faces em quadros estáticos, enquanto técnicas de rastreio como o Camshift são capazes de acompanhar o conteúdo da janela de detecção através dos diferentes quadros de uma sequencia de imagens.

Em conjunto, observamos como diferentes sensores e representações de imagens podem ser utilizados para extrair informações diferentes de uma cena. O uso de uma câmera de profundidade nos permite estimar mapas bidimensionais contendo uma estimativa da distância entre a câmera e o objeto em cada pixel da imagem.

Destas duas técnicas, vimos como podemos elaborar um mecanismo de segmentação dos membros superiores do corpo humano combinando informação disponível em imagens coloridas (a localização da face) e de profundidade (a localização de objetos e regiões que se localizem a frente à face).

No próximo capítulo, veremos será possível *interpretar* o que está sendo segmentado por este algoritmo, de maneira a obtermos, por exemplo, uma estimativa do que o está sendo articulado no espaço de sinalização em termos das unidades básicas do sinal identificadas no Capítulo 3.

Capítulo 5

Reconhecimento de padrões

"Essentially, all models are wrong, but some are useful." — George E. P. Box, 1987

O RECONHECIMENTO DE padrões desempenha papel significante na área de visão computacional. Aliado ao processamento de imagens, as técnicas de reconhecimento de padrões (e, por conseguinte, a inteligência artificial e o aprendizado de máquina) são as principais responsáveis por decodificar as imagens registradas por sensores em símbolos ou sequências de símbolos que possam ser interpretados por uma máquina. Uma vez transformados em símbolos, a máquina pode então tomar decisões adequadas como responder um respectivo comando ao usuário. Neste capítulo, apresentaremos técnicas gerais de aprendizado de máquina e reconhecimento de padrões que encontraram aplicações imediatas na visão computacional e serão de suma importância na execução desta pesquisa.

5.1 Redes Neurais Artificiais

Desde sua origem, muito da pesquisa e aplicações das redes neurais tem sido motivada por forte inspiração biológica. No começo da década de 40, McCullogh e Pitts (1943) publicaram um trabalho descrevendo seu modelo simplificado de neurônio criado com circuitos eletrônicos, ilustrado na Figura 44. Baseado neste modelo de neurônio artificial, Rosenblatt (1958), neurobiólogo na universidade de Cornell, desenvolveu o algoritmo *Perceptron* enquanto pesquisava o sistema neurovisual em moscas. Nos anos seguintes, Werbos (1974) desenvolveu melhorias ao modelo *Perceptron* estendendo-o ao modelo de algoritmo hoje conhecido como retropropagação de erro - que curiosamente não se popularizou até ser redescoberto em 1986 por MacClelland e Rumelhart (1986).

Neste curto resumo sobre a origem da área, pode-se notar o quanto muitos dos pesquisadores que a inauguraram estavam mais interessados nas características e implicações biológicas e psicológicas do aprendizado do que em sua caracterização e formulação matemática, o que acabou resultando em uma fraca análise e sustentação teórica. Por anos, redes neurais foram vistas como modelos *caixa-preta* em que o aprendizado ocorria por razões pouco compreendidas. Apenas recentemente as redes neurais começaram a ser estudadas de um ponto de vista mais formal, contando com completos tratamentos matemáticos e estatísticos. Explorando este novo ponto de vista, tornou-se clara e imediata a possibilidade de aplicação de um enorme ferramental oriundo da teoria de otimização matemática e do aprendizado estatístico ao invés de técnicas inspiradas na biologia.

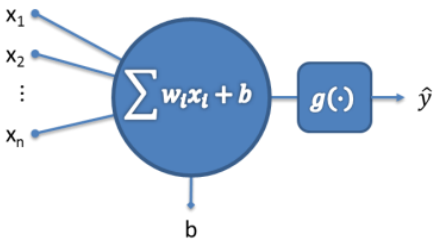

Figura 44. Representação de um neurônio artificial com uma função de ativação não linear. Note que o modelo Perceptron é um caso especial quando g(x) = sgn(x).

A falta desta formação teórica mais sólida talvez tenha sido a causa do grande recesso em pesquisa na área ocorrida a partir da década de 70. No final da década de 60, Minsky e Papert (1969) publicaram um dos resultados de sua análise teórica do algoritmo *Perceptron*, provando que a aplicabilidade do modelo de único neurônio se restringia apenas a problemas linearmente separáveis. No entanto, assumiu-se erroneamente que este resultado também se aplicava a redes neurais de múltiplas camadas, o que acabou atrasando a pesquisa na área por muitos anos. Apenas muito mais tarde, Cybenko (1989) provou que redes neurais com funções de ativação sigmoidal eram aproximadores universais, e finalmente Hornik (1991) verificou que não era a escolha da função de ativação em si que garantia este resultado, mas sim a disposição em camadas em que suas unidades neuronais eram dispostas. Este resultado ficou conhecido sobre o nome de *Teorema da Aproximação Universal* e resgatou muito do interesse em pesquisa na área.

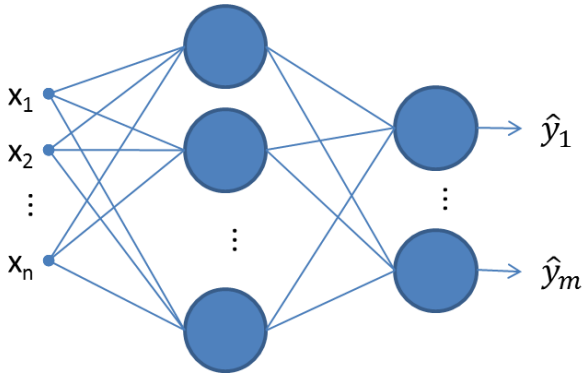

Figura 45. Modelo de rede neural feed-forward para n entradas e m saídas com duas camadas escondidas, sendo uma camada intermediaria e uma de saída.

Em termos gerais, uma rede neural pode ser vista como uma combinação de unidades simples conectadas entre si numa hierarquia de camadas (Figura 45). Estas unidades simples visam modelar o funcionamento de um neurônio biológico como o modelo de McCullogh e Pitts. A arquitetura mais comum utilizada para dispor as camadas em redes neurais é a arquitetura *feed-forward*, sendo que redes adotando esta arquitetura também são conhecidas como *Perceptrons multicamadas*. No entanto, o uso do nome *Perceptron* é na verdade impróprio, já que o modelo *Perceptron* emprega uma função não linear e não contínua (BISHOP, 2007, p. 226), diferente do modelo de redes neurais baseadas em funções sigmoidais que apresentam funções não lineares, mas contínuas. De fato, o modelo utilizando funções de ativação sigmoidais pode ser visto como, na verdade, uma rede de regressões logísticas. Apesar do modelo de rede de única camada intermediária ter sido provado ser um aproximador universal, redes com duas ou mais camadas intermediárias muitas vezes se mostraram mais eficientes (com menos unidades ocultas para o aprendizado de um problema) do que redes de única camada (HASSOUN, 1995).

Apesar da fundamentação biológica e de sua característica paralela e multicamada, uma rede neural pode ser simplesmente vista como uma função $f: \mathbb{R}^n \rightarrow \mathbb{Y}$. O conjunto $\mathbb{Y}$ denota o conjunto de possíveis saídas da rede neural, em que cada elemento $\boldsymbol{y} \in \mathbb{Y}$ tem a forma $\boldsymbol{y} = \langle y_1, \ldots, y_m \rangle$. Por sua vez, cada um dos elementos $y_i \in \boldsymbol{y}$ do vetor de saídas está restrito a um intervalo em particular $[a; b]$. Este intervalo dos elementos de saída é em geral definido pela função de ativação escolhida para os neurônios de saída da rede. No caso da função sigmoidal, este intervalo é o intervalo $[0; 1]$; para o caso da função sigmoidal bipolar este intervalo é $[-1; +1]$.

Esta visão da rede neural como uma simples função dá margem a aplicação de métodos bem conhecidos de otimização de funções para realizar seu aprendizado.

Do ponto de vista do aprendizado, pode-se dizer que a função f é pertencente a uma classe de funções $\mathcal{F}$ com alguma forma específica ditada pela escolha da arquitetura e funções de ativação da rede. Além de sua arquitetura, estas funções também são parametrizadas de acordo com possíveis vetores de peso $\boldsymbol{\theta} \in \mathbb{R}^w$, em que w é número total de parâmetros na rede, contando pesos e valores de viés (*biases*). Podemos representar esta parametrização como

$$\mathrm{f}(\mathbf{x};\ \boldsymbol{\theta}) = \hat{\mathbf{y}} \tag{5.1}$$

ou mesmo $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = \hat{\boldsymbol{y}}$.

O aprendizado então consiste em achar o melhor vetor de pesos $\boldsymbol{\theta}$ tal que, quando o vetor de entradas $\boldsymbol{x}$ seja fornecido à rede, suas correspondentes saídas $\hat{\boldsymbol{y}}$ sejam as mais próximas possíveis de algum valor esperado $\boldsymbol{y}$. Num contexto matemático, este problema pode ser moldado como um problema em que se busca o máximo ou mínimo de algum critério de otimização, como o mínimo erro quadrático, a área máxima sob uma curva ROC, ou a máxima margem de separação entre os dados (no caso de um problema estrito de classificação).

Muitos dos algoritmos mais populares para o aprendizado de redes neurais são baseados na minimização do gradiente de erro. Exemplos comuns desta classe de algoritmos são dados pelo algoritmo de Gradiente Descendente Estocástico (*Stochastic Gradient Descent*, SGD) e as muitas variações do método de Gauss-Newton. É possível, no entanto, aplicar as mais diversas técnicas para se obter o vetor $\boldsymbol{\theta}$ mais adequado; um dos mais interessantes exemplos é a aplicação de algoritmos genéticos (MONTANA e DAVIS, 1989) por combinar duas abordagens biológicas num mesmo problema.

No entanto, deixando-se de lado a motivação biológica e focando mais em objetividade científica na resolução de um problema em particular, a escolha de um método de aprendizado adequado para redes neurais é de importância crucial. O problema de aprendizado em redes neurais é frequentemente um problema mal condicionado (SAARINEN, BRAMLEY e CYBENKO, 1993). Na presença de mal condicionamento, métodos de primeira ordem como os métodos de gradiente descendente podem se tornar severamente lentos. Em outros casos, a escolha de um tamanho de passo muito largo pode levar a divergência, e um passo muito pequeno a um treinamento muito vagaroso. É possível notar que o melhor tamanho de passo nos métodos de primeira ordem é específico para cada peso da rede. Nas seções seguintes, apresentaremos alguns métodos para se evitar estes problemas.

5.1.1 O algoritmo de Levenberg-Marquardt

Uma das melhores maneiras de se obter uma rápida convergência no aprendizado é utilizar qualquer informação de segunda ordem que esteja disponível para guiar a minimização do gradiente. Um exemplo clássico de algoritmo de segunda ordem é o método de Newton. Quando o método de Newton converge, converge quadraticamente. No entanto, deve-se notar que computar a matriz Hessiana $\boldsymbol{H}$ de derivadas parciais de segunda ordem requerida pelo algoritmo é extremamente custoso e muitas vezes proibitivo, tanto em termos de tempo quanto em termos de memória.

Quando o problema de otimização puder ser expresso como um problema de mínimos quadrados, que é o caso quando utilizamos o critério de mínimo erro quadrático como função de custo durante o aprendizado de redes neurais, então se torna possível utilizar o método de Gauss-Newton para caminhar no gradiente sem que seja necessária a computação de $\boldsymbol{H}$. No entanto, novamente devemos observar que o método de Newton nem sempre é convergente, podendo divergir caso a solução inicial esteja muito longe de um mínimo local.

O algoritmo de Levenberg-Marquardt (*Levenberg Marquardt Algorithm*, LMA) descrito em (MARQUARDT, 1963) e discutido em (HAGAN e MENHAJ, 1994) é uma alternativa popular ao método de Gauss-Newton. O LMA é capaz de prover um balanço entre o método de Newton e a descida de gradiente, sendo consequentemente mais robusto. Assim como o método de Gauss-Newton, o algoritmo utiliza uma aproximação para a matriz $\boldsymbol{H}$ utilizando-se apenas informação disponível na matriz Jacobiana $\boldsymbol{J}$ de derivadas parciais de primeira ordem. Cada iteração realizada pelo LMA consiste em resolver a equação

$$(\mathbf{J}^{\mathrm{T}}\mathbf{J} + \lambda\mathbf{I})\boldsymbol{\delta} = \mathbf{J}^{\mathrm{T}}\,\mathbf{e} \tag{5.2}$$

em que J é a matriz Jacobiana para o sistema, λ é o fator de amortecimento de Levenberg, $\boldsymbol{\delta}$ é o vetor de ajuste dos pesos (que desejamos encontrar) e $\boldsymbol{e}$ é o vetor de erros quadráticos contendo os erros de aproximação para cada vetor de entradas utilizado no treinamento da rede. A matriz Jacobiana é a matriz de todas derivadas de primeira ordem de uma função vetorial. Considerando um conjunto de n dados de treinamento $\boldsymbol{\mathcal{D}} = \{\boldsymbol{x_i}, \boldsymbol{y_i}\}_i^n$ e um vetor de parâmetros $\boldsymbol{\theta} = \langle\theta_1, \ldots, \theta_w\rangle$, a matriz Jacobiana é a matriz $n \times w$ definida por

$$\mathbf{J} = \begin{pmatrix} \frac{\partial \mathrm{f}(\mathbf{x_1};\, \boldsymbol{\theta})}{\partial \theta_1} & \cdots & \frac{\partial \mathrm{f}(\mathbf{x_1};\, \boldsymbol{\theta})}{\partial \theta_w} \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathrm{f}(\mathbf{x_n};\, \boldsymbol{\theta})}{\partial \theta_1} & \cdots & \frac{\partial \mathrm{f}(\mathbf{x_n};\, \boldsymbol{\theta})}{\partial \theta_w} \end{pmatrix} \tag{5.3}$$

No caso geral de otimização de funções sem uma forma conhecida, ou em que o vetor gradiente em relação a uma amostra $\boldsymbol{x_i}$ seja desconhecido ou difícil de ser calculado de maneira direta, a Jacobiana pode ser aproximada utilizando-se métodos de diferenças finitas. No entanto, para o caso de redes neurais, o vetor gradiente e consequentemente a matriz Jacobiana $\boldsymbol{J}$ pode ser calculada de maneira eficiente utilizando-se a regra da cadeia através do algoritmo de retropropagação.

Na equação (5.2) o ajuste dos pesos $\boldsymbol{\delta}$ denota o quanto é necessário modificar o vetor de parâmetros $\boldsymbol{\theta}$ para obter uma possível melhor solução, enquanto que a matriz $\boldsymbol{J^T J}$ consiste em uma aproximação local para a matriz Hessiana $\boldsymbol{H}$, de maneira que $\boldsymbol{H} \cong \boldsymbol{J^T J}$. No entanto, nesta aproximação local supõe que f seja linear em $\boldsymbol{\theta}$, o que, especialmente no caso de redes neurais, nem sempre é verdade. Para circunverter esta dificuldade, o algoritmo confia nesta suposição apenas quando esta aproximação linear é razoável: o fator de amortecimento λ controla o quanto esta aproximação deve ser utilizada em cada iteração, guiando o processo de otimização. Se o erro a cada iteração está se reduzindo rapidamente, uma diminuição do valor de λ traz o algoritmo mais perto do método de Gauss-Newton; já se uma iteração provê uma redução insuficiente no erro, λ pode ser incrementado para trazer o algoritmo mais perto da direção do gradiente descendente.

Adicionalmente, o uso do método de Levenberg-Marquardt possibilita a fácil realização de técnicas de regularização como a regularização Bayesiana[12]. Para circunverter o problema de interpolar dados ruidosos, MacKay (1992) propôs um *framework bayesiano* que pode ser aplicado diretamente ao problema de aprendizado de redes neurais. Seu framework também pode ser utilizado para se estimar o numero efetivo de parâmetros utilizados pelo modelo – neste caso, o número de pesos da rede realmente requeridos para se resolver determinado problema. A regularização bayesiana expande a função objetivo para buscar não somente pelo menor erro, mas pelo menor erro utilizando pesos mínimos. Funciona através da introdução de dois hiperparâmetros bayesianos, α e β, dizendo qual direção (erro mínimo ou pesos

[12] A regularização é uma das formas mais comuns de se obter controle da capacidade. A importância de se impor controle da capacidade a um modelo é defendida na teoria de aprendizado estatístico de Vapnik, apresentada com mais detalhes na seção 5.2.2.

mínimos) o processo de aprendizado precisa buscar. Quando aplicados a algoritmos iterativos como o LMA, estes hiperparâmetros são atualizados ao final de cada iteração. No entanto, de acordo com Poland (2001), a escolha mais popular de atualização falha em produzir iterações robustas caso existam apenas poucos dados de treinamento disponíveis.

Em sua forma direta, apesar de utilizar apenas uma aproximação para a matriz $\boldsymbol{H}$, o custo de memória do método de Levenberg-Marquardt ainda pode ser proibitivo. Esta aproximação ocupará espaço em memória proporcional ao quadrado do número w de parâmetros na rede, e a inversão desta matriz, cujo cômputo pode ser necessário múltiplas vezes a cada iteração, também requererá alta quantidade de processamento. Algumas técnicas podem ser utilizadas para otimizar o uso da memória, como o uso da decomposição de Cholesky para evitar a inversão da matriz $\boldsymbol{H}$; e o cômputo gradual da matriz $\boldsymbol{H}$ através de múltiplas Jacobianas menores. O cálculo de elementos específicos da matriz inversa, como seu traço, também pode ser computado de maneira eficiente e sem utilizar memória adicional explorando certas características da decomposição de Cholesky, como demonstrado em (BJÖRCK, 1996, p. 119-120).

5.1.2 O algoritmo de Retropropagação Resiliente

A maior vantagem do algoritmo de Levenberg-Marquardt é também seu ponto fraco. Por aproximar a informação de segunda ordem através da construção de uma aproximação para a matriz Hessiana $\boldsymbol{H}$, o método requer quantidade de memória suficiente para conter esta matriz. No entanto, como citado na seção anterior, a matriz $\boldsymbol{H}$ é uma matriz $w \times w$, escalando em memória como $\boldsymbol{O}(w^2)$ em relação ao número w de parâmetros da rede, e cuja computação acaba requerendo alta quantidade de memória e processamento. Caso a rede neural apresente um alto número de parâmetros, sua computação e armazenamento podem tornar-se intratáveis e o método deixa de ser uma alternativa viável.

Para circunverter este problema e ainda assim apresentar convergência acelerada, Riedmiller (1994) desenvolveu o algoritmo de Retropropagação Resiliente (*Resilient Backpropagation*, Rprop), posteriormente modificado por Igel e Hüsken (2000). O algoritmo de aprendizado Rprop é um dos métodos de aprendizado mais rápidos restritos a informação de apenas primeira ordem. O princípio básico de sua operação é eliminar a influência potencialmente negativa da magnitude do gradien-

te durante o passo de otimização. Diferentemente de métodos como o algoritmo de Gradiente Descendente, em que o passo é sempre proporcional ao vetor gradiente, no Rprop apenas a direção do gradiente é mantida, e o passo atualizado de acordo com

$$\Delta w_{ij} = \begin{cases} -\Delta_i^{(t)}, & se \frac{\partial E}{\partial \theta_i}^{(t)} > 0 \\ +\Delta_i^{(t)}, & se \frac{\partial E}{\partial \theta_i}^{(t)} < 0 \\ 0, & caso\ restante \end{cases} \tag{5.4}$$

em que $\frac{\partial E}{\partial \theta_i}^{(t)}$ denota o gradiente acumulado sobre todos os padrões de aprendizado durante a iteração t. Os valores de atualização $\Delta_{ij}^{(t)}$ representam um fator de aprendizado adaptativo relacionado com cada peso θ_i da rede, que é modificado de acordo com a fórmula

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^+ * \Delta_i^{(t-1)}, & se \frac{\partial E}{\partial \theta_i}^{(t-1)} * \frac{\partial E}{\partial \theta_i}^{(t)} > 0 \\ \eta^- * \Delta_i^{(t-1)}, & se \frac{\partial E}{\partial \theta_i}^{(t-1)} * \frac{\partial E}{\partial \theta_i}^{(t)} < 0 \\ \Delta_i^{(t-1)}, & caso\ restante \end{cases} \tag{5.5}$$

em que $0 < \eta^- < 1 < \eta^+$.

Aqui, as atualizações são controladas por fatores de incremento de decremento η^+ e η^-, respectivamente. Para evitar que as atualizações aumentem ou diminuam excessivamente, impõem-se o limite de que os valores de $\Delta_i^{(t)}$ estejam entre Δ_{max} e Δ_{min}. A principal vantagem do algoritmo de Rprop é apresentar taxas de convergência rápidas sem utilizar informações de segunda ordem, evitando assim utilizar altas quantidades de memória. O método escala como $\boldsymbol{O}(w)$ em relação ao número w de pesos e vieses da rede. A seguir apresentamos o algoritmo de Rprop em pseudocódigo similar ao apresentado em (RIEDMILLER, 1994).

Algoritmo. Retropropagação Resiliente

Inicialização

$\Delta_i^{(t)} \leftarrow 0,\ \frac{\partial E}{\partial \theta_i}^{(t-1)} \leftarrow 0, \quad i, \dots, w$

Repita

Calcule o vetor gradiente $\frac{\partial E}{\partial \theta_{ij}}^{(t)}$

Para todos os pesos e vieses θ_i da rede, **faça:**

Se $\frac{\partial E}{\partial \theta_i}^{(t-1)} \cdot \frac{\partial E}{\partial \theta_i}^{(t)} > 0$ **então:**

$\Delta_i^{(t)} \leftarrow \min\left(\Delta_i^{(t-1)} * \eta^+, \Delta_{max}\right)$

$\delta_i^{(t)} \leftarrow -\text{sign}\left(\frac{\partial E}{\partial \theta_i}^{(t)}\right) * \Delta_i^{(t)}$

$\theta_i^{(t+1)} \leftarrow \theta_i^{(t)} + \delta_i^{(t)}$

$\frac{\partial E}{\partial \theta_i}^{(t+1)} \leftarrow \frac{\partial E}{\partial \theta_i}^{(t)}$

Senão, se $\frac{\partial E}{\partial \theta_i}^{(t-1)} \cdot \frac{\partial E}{\partial \theta_i}^{(t)} < 0$ **então:**

$\Delta_i^{(t)} \leftarrow \max\left(\Delta_i^{(t-1)} * \eta^-, \Delta_{min}\right)$

$\frac{\partial E}{\partial \theta_i}^{(t-1)} \leftarrow 0$

Senão,

$\delta_i^{(t)} \leftarrow -\text{sign}\left(\frac{\partial E}{\partial \theta_i}^{(t)}\right) * \Delta_i^{(t)}$

$\theta_i^{(t+1)} \leftarrow \theta_i^{(t)} + \delta_i^{(t)}$

$\frac{\partial E}{\partial \theta_i}^{(t+1)} \leftarrow \frac{\partial E}{\partial \theta_i}^{(t)}$

até (convergência)

Figura 46. O algoritmo de Retropropagação Resiliente (RIEDMILLER, 1994).

5.2 Máquinas de Vetores de Suporte

Apesar de sua maior robustez, os métodos de aprendizado apresentados na seção anterior apresentam severas desvantagens, em parte devido ao fato da função objetivo a ser otimizada no aprendizado de ANNs apresentar múltiplos mínimos locais. De maneira geral, muitos dos algoritmos para aprendizado de ANNs sequer apresentam garantias de convergência, e em prática requerem múltiplas reinicializações aleatórias dos pesos sinápticos para garantir que um bom aprendizado (um mínimo local próximo do global) seja atingido. Mesmo assim, não há como garantir que este mínimo local seja o melhor possível, e como bem elucida (VAPNIK, 1998, p. 714), muitos pesquisadores consideram as aplicações de redes neurais em problemas da vida real como sendo "mais arte do que ciência".

Em contraste com as ANNs, as Máquinas de Vetores de Suporte (*Support Vector Machines*, SVM) apresentam muitas características ótimas, tanto em termos de decisão quanto de aprendizado (VAPNIK, 1998). Seu aprendizado é convexo, o que significa que apresentam um mínimo local coincidindo com o mínimo global. São esparsas, no sentido que conseguem ajustar seu número de parâmetros automaticamente. Fornecem certa intuição na análise de seus resultados, já que selecionam alguns exemplos do conjunto de treinamento para traçar sua superfície de decisão (CRISTIANINI e SHAWE-TAYLOR, 2000). No entanto, uma dificuldade é que todo o conjunto de dados de treinamento deve estar disponível antes de o aprendizado começar. Técnicas para o aprendizado *online* existem, porém não são usuais ou apresentam dificuldades por si só. Outra desvantagem está em sua generalização para problemas de múltiplas classes, visto que toda sua formulação tem origem em problemas binários de classificação.

Apesar destas dificuldades, as SVMs tem se mostrado muito robustas em muitas aplicações, como no reconhecimento óptico de caracteres (LECUN, JACKEL, *et al.*, 1995; LIU, NAKASHIMA, *et al.*, 2003), categorização de textos (JOACHIMS, 1998), detecção de faces (OSUNA, FREUND e GIROSIT, 1997) e expressões faciais em vídeo (MICHEL e EL KALIOUBY, 2003). É possível explicar este sucesso empírico considerando que a SVM é capaz de aproximar assintoticamente a classificação ótima de Bayes de uma maneira muito eficiente, sem que seja necessário estimar a função de probabilidade condicional do problema de decisão, conforme argumentado e demonstrado em (LIN, 2002) e (LIN, LEE e WAHBA, 2002).

Para compreender melhor as principais características e demais ideias que sustentam a fundamentação teórica por trás das SVMs, podemos apresentar uma segunda visão do algoritmo *Perceptron* mencionado anteriormente neste capítulo, o que faremos a seguir.

5.2.1 Do Perceptron às Máquinas Vetores de Suporte

O algoritmo *Perceptron* (ROSENBLATT, 1958) foi um dos primeiros algoritmos de treinamento neural, apresentando severas limitações. No entanto, sua simplicidade serve de base para se explorar conceitos muito úteis. No algoritmo *Perceptron*, o ajuste dos pesos iniciais funciona através da adição ou subtração de valores proporcionais aos exemplos de treinamento classificados erroneamente, de forma a se identificar um hiperplano (MINSKY, 1961) no espaço de entradas separando os exemplos de treinamento. Sua função de decisão tem a forma

$$h(\mathbf{x}) = \text{sgn}(\boldsymbol{\theta} \cdot \mathbf{x} + b) \tag{5.6}$$

em que $sgn(\cdot)$ denota a função sinal, com a definição adicional de que $sgn(0) = 1$. A Figura 47 exibe o algoritmo Perceptron em sua forma original, mostrando como os pesos são ajustados sempre que há um erro na classificação de uma amostra.

Algoritmo. Perceptron em sua forma primal
Dado um conjunto de ℓ dados linearmente separáveis S e uma taxa de aprendizado $\eta \in \mathbb{R}^+$
$\theta_0 \leftarrow 0, b_0 \leftarrow 0,$
$k \leftarrow 0$
$R \leftarrow \max_{1 \le i \le l} \|\mathbf{x_i}\|$
Repita
  Para $i = 1$ **até** ℓ
    Se $y_i(\theta_k \cdot \mathbf{x_i}) + b_k \le 0$ **então**
      $\theta_{k+1} \leftarrow \theta_k + \eta\, y_i\, x_i$
      $b_{k+1} \leftarrow b_k + \eta\, y_i\, R^2$
      $k \leftarrow k + 1$
    Fim-se
  Fim-para
Até que nenhum erro seja cometido dentro do laço para
Retorne (θ_k, b_k) em que k é o número de erros cometidos durante o aprendizado.

Figura 47. O algoritmo Perceptron em sua forma primal, traduzido e adaptado de (CRISTIANINI e SHAWE-TAYLOR, 2000).

Pode-se notar que, assumindo que o vetor inicial de pesos seja o vetor zero, pode-se concluir que a hipótese final para os pesos $\boldsymbol{\theta}$ será dada por uma combinação linear dos ℓ pontos de treinamento $\boldsymbol{x_i}$ e associados rótulos $y_i \in \{-1, 1\}$. É fácil verificar essa afirmativa observando que o vetor de pesos será sempre ajustado por um valor proporcional a uma amostra específica pertencente ao conjunto de dados a cada iteração. Se combinarmos todas estas atualizações em um único passo, cada amostra estará associada a um escalar $\boldsymbol{\alpha_i}$ que determina o peso desta amostra no decorrer de todas as atualizações. Assim, pode-se expressar o vetor de pesos final na forma (CRISTIANINI e SHAWE-TAYLOR, 2000)

$$\boldsymbol{\theta} = \sum_{i=1}^{l} \alpha_i \, y_i \, \mathbf{x_i} \, . \tag{5.7}$$

Como o sinal do valor proporcional de ajuste será dado pelo rótulo y_i, os valores $\boldsymbol{\alpha_i}$ deverão ser valores positivos proporcionais ao número de vezes que uma classificação errada de $\boldsymbol{x_i}$ causou o reajuste do vetor de pesos. Uma vez que o conjunto de treinamento S seja fixado, é possível visualizar $\boldsymbol{\alpha}$ como a representação dual da hipótese em um espaço de coordenadas diferente.

Algoritmo. Perceptron em sua forma dual
Dado um conjunto de ℓ dados linearmente separáveis S
$\boldsymbol{\alpha} \leftarrow 0, b \leftarrow 0,$
$R \leftarrow \max_{1 \le i \le l} \|\mathbf{x_i}\|$
Repita
  Para $i = 1$ **até** ℓ
    Se $y_i \left(\left(\sum_{j=1}^{\ell} \alpha_j \, y_j \, \mathbf{x_j} \right) \cdot \mathbf{x_i} \right) + b_k \le 0$ **então**
      $\alpha_i \leftarrow \alpha_i + 1$
      $b_i \leftarrow b_i + y_i R^2$
    Fim-se
  Fim-para
Até que nenhum erro seja cometido dentro do laço para
Retorne $(\boldsymbol{\alpha}, b)$ em que $\boldsymbol{\alpha}$ pode ser utilizado para formar $\boldsymbol{\theta}$.

Figura 48. O algoritmo Perceptron em sua forma dual, traduzido e adaptado de (CRISTIANINI e SHAWE-TAYLOR, 2000).

Substituindo-se a fórmula anterior na função de decisão, pode-se chegar a sua forma dual, dada por

$$\begin{aligned} h(\mathbf{x}) = \operatorname{sgn}(\boldsymbol{\theta} \cdot \mathbf{x} + b) &= \operatorname{sgn}\left(\left(\sum_{i=1}^{\ell} \alpha_i y_i \mathbf{x_i}\right) \cdot \mathbf{x} + b\right) \\ &= \operatorname{sgn}\left(\sum_{i=1}^{\ell} \alpha_i y_i (\mathbf{x_i} \cdot \mathbf{x}) + b\right). \end{aligned} \tag{5.8}$$

A forma dual é um dos conceitos chaves para se compreender a formulação das SVMs. Esta reformulação também possibilita expressar a decisão utilizando-se apenas produtos internos, o que é requisito básico para aplicação do truque do *kernel*. Note que a compreensão de que o algoritmo *Perceptron* empregava a simples busca de um hiperplano deu margens para que se empregassem técnicas muito superiores ao primeiro algoritmo sugerido por Rosenblatt (1958).

5.2.2 O framework de aprendizado de Vapnik

Métodos e algoritmos tradicionais do aprendizado de máquina e reconhecimento de padrões, em particular os métodos gerativos, são geralmente baseados na estimação de parâmetros de alguma distribuição de probabilidade (como, por exemplo, pelo critério de *máxima verossimilhança*). A popularidade da abordagem paramétrica para estimação de densidades deve-se muito a Fisher, que simplificou o problema de se estimar uma densidade ao problema de se estimar poucos parâmetros de funções de densidade (FISHER, 1922). Parafraseando Vapnik, esta simplificação mostrou-se muito útil em sua época, em que recursos computacionais se resumiam a lápis e papel, e simplicidade de cálculos era necessária.

No entanto, com o advento da era da computação e da necessidade de se resolver problemas de dimensionalidade muito maiores, encontrou-se um problema ao aplicar os métodos de Fisher em problemas de alta-dimensionalidade. Descobriu-se que, utilizando-se apenas o framework paramétrico de Fisher, era impossível circunverter o "*mal da dimensionalidade*" (VAPNIK, 1998, p. 683). Uma das mais importantes conquistas da teoria de Vapnik fora a descoberta de que a habilidade de generalização da máquina de aprendizado depende na capacidade do conjunto de funções implementadas pela máquina, que é diferente de seu número de parâmetros livres. Isto permitiu que se superasse a barreira imposta pela maldição da dimensi-

onalidade, como foi conhecido o problema de se trabalhar com métodos clássicos (ou paramétricos) em altas dimensões.

Um dos conceitos básicos de sua teoria é a dimensão de Vapnik–Chervonenkis (dimensão VC) para medida de capacidade de um modelo ou classe de funções, exemplificado na Figura 49. Este conceito pode ser visto como uma generalização do conceito de capacidade primeiro introduzido por Cover (1965), sendo uma caracterização combinatória da diversidade de funções que podem ser aprendidas por uma máquina de aprendizado. A dimensão de Vapnik-Chervonenkis de uma classe de funções $\mathcal{F}$ definidas sobre um espaço de instâncias X é o tamanho do maior subconjunto de X *partido* por $\mathcal{F}$. Um conjunto de amostras S é dito ser *partido* por uma classe de funções $\mathcal{F}$ se e somente se, para cada possível rotulação das amostras em S existe alguma função em $\mathcal{F}$ que perfeitamente classifique estas amostras (VAPNIK, 1998, p. 147). Um exemplo do cálculo da dimensão VC para hiperplanos em $\mathbb{R}^2$ é exibido na Figura 49 e sua legenda.

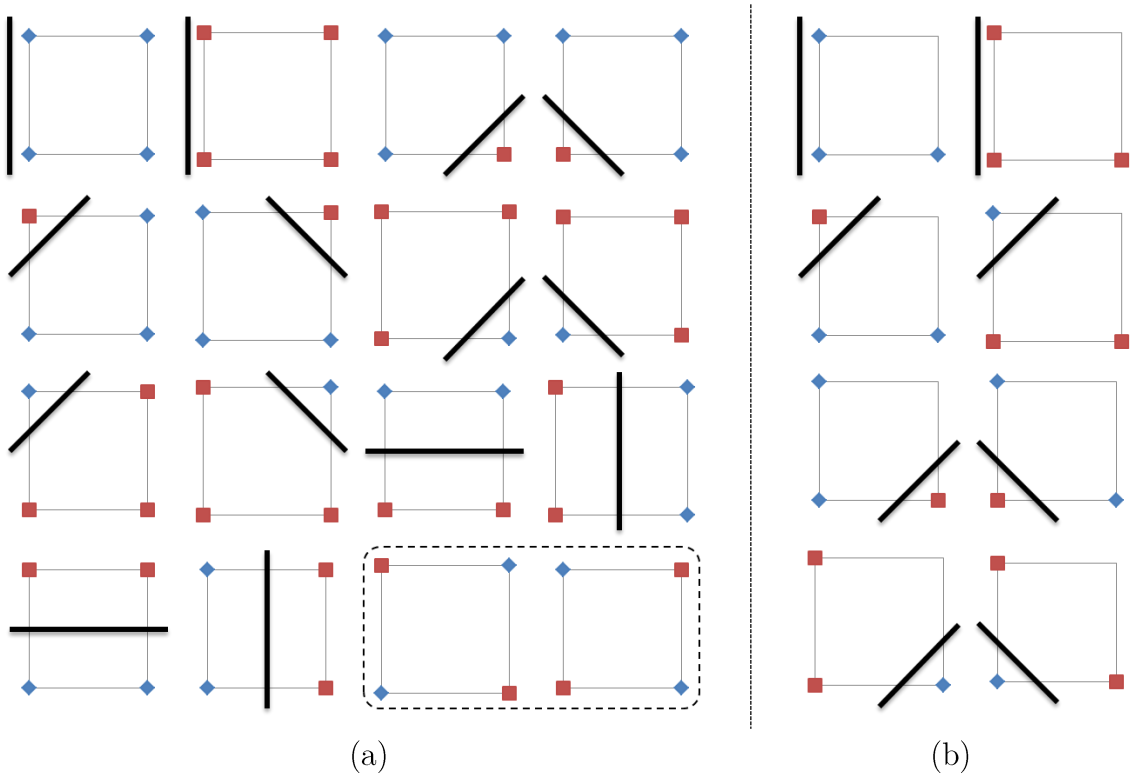

Figura 49. Exemplificação da dimensão VC para a classe de funções de separação dada por hiperplanos (separadores lineares). Figura (a) mostra que um conjunto de quatro pontos não pode ser partido por hiperplanos em um espaço $\mathbb{R}^2$ (indicados pelo retângulo tracejado). Em contrapartida, figura (b) mostra que um subconjunto de três pontos pode de fato ser partido por hiperplanos. A dimensão VC de hiperplanos em $\mathbb{R}^2$ é, portanto, igual a 3. Em geral, a dimensão VC de hiperplanos em $\mathbb{R}^d$ é d+1.

Uma observação fundamental realizada por Vapnik (1998, p. 33) é que, se a dimensão de VC de um conjunto de funções é infinita, então a situação de não falseabilidade descrita por Karl Popper (1959) impede[13] que qualquer generalização se torne possível. Desta maneira, modelos de menor dimensão VC seriam sempre preferíveis. Para obter um controle sobre a dimensão VC de um modelo, sua teoria faz uso extensivo do controle da capacidade, sem que este controle da capacidade dependa da dimensionalidade dos dados.

Utilizando esta nova teoria, pode-se mostrar que máquinas com um elevado número de parâmetros também podem apresentar elevado grau de generalização, o que seria contra intuitivo seguindo-se a navalha de Occam original. A capacidade de generalização depende apenas da capacidade, e não do número de parâmetros livres do modelo. Em virtude desta característica, Vapnik propôs a modificação da navalha de Occam original

> *A explicação mais simples é a melhor.*

para

> *A explicação dada pela máquina com menor capacidade (dimensão VC) é a melhor.*

Baseado neste objetivo, Vapnik definiu um principio de indução a ser conhecido como Minimização do Risco Estrutural (*Structural Risk Minimization*, SRM). O Princípio Primordial da SRM diz que,

> Se há apenas um conjunto de informação restrito para a resolução de algum problema, deve-se resolver o problema direto e nunca resolver um problema mais geral como um passo intermediário. É possível que a informação disponível seja suficiente para uma solução direta, mas não para resolver um problema intermediário mais geral (VAPNIK, 1998, p. 12).

Conforme prossegue Vapnik em sua argumentação, pode-se notar que este princípio vai diretamente contra a abordagem da estatística paramétrica convencional. Para estimarmos uma densidade condicional, por exemplo, poderíamos primeiro estimar duas densidades, $p(x, y)$ e $p(y)$, e então computar sua razão $p(x, y)/p(y) = p(y|x)$. No entanto, pode-se notar (VAPNIK, 1998) que a estima-

[13] Note que Popper também tentou definir sua própria dimensão para medida de poder de um modelo. No entanto, existem divergências nas duas abordagens que talvez possam ser explicadas por enganos na tentativa de Popper em definir matematicamente os conceitos filosóficos que apresentava (CORFIELD, VAPNIK, *et al.*, 2005).

ção de densidades é um problema geral que pode ser utilizado para a resolução dos mais variados problemas (é, por exemplo, o primeiro passo a ser executado na estimação por máxima verossimilhança). Desta forma, conhecer uma densidade possibilita resolver uma grande quantidade de problemas, muito além do que estamos realmente interessados, que é apenas o problema de classificação. Complicando ainda mais esta situação, a estimação de densidades é um problema difícil e mal condicionado que requer um alto número de observações para ser resolvido adequadamente. Neste caso, de acordo com a SRM, nota-se que a estimação de densidades como passo intermediário a classificação é algo que deve ser evitado.

Um dos frutos mais produtivos da teoria de Vapnik é representado pela classe de algoritmos de aprendizados dada pelas Máquinas de Vetores de Suporte. Aplicando todos os princípios da Teoria de Aprendizado Estatístico, as SVMs foram um dos primeiros métodos classificadores não suscetíveis ao mal da dimensionalidade; ao mesmo tempo em que incorporam mecanismos para o controle da capacidade através do controle da margem de separação dos dados.

5.2.3 Classificadores de máxima margem

Assim como no caso do algoritmo *Perceptron*, máquinas de decisão linear tentam encontrar um hiperplano que divida o espaço de entradas em duas regiões. Pontos residindo em cada região caracterizariam dados de classes diferentes. A equação de um hiperplano é dada por

$$f(\mathbf{x}) = \boldsymbol{\theta} \cdot \mathbf{x} + b = 0 \tag{5.9}$$

e a decisão em favor de uma determinada classe é dada de acordo com

$$h(\mathbf{x}) = \mathrm{sgn}\big(f(\mathbf{x})\big) = \mathrm{sgn}(\boldsymbol{\theta} \cdot \mathbf{x} + b) \tag{5.10}$$

com a definição adicional de que $sgn(0) = 1$. Sob essas circunstâncias, podemos ver que esta é exatamente a formulação do modelo *Perceptron*. No entanto, Vapnik considerou encontrar não apenas um hiperplano que divida os dados, mas o hiperplano que divida os dados tal que a margem de separação deste hiperplano seja máxima. A margem (distância do hiperplano aos pontos) é dada por $1/\|\boldsymbol{\theta}\|^2$ e po-

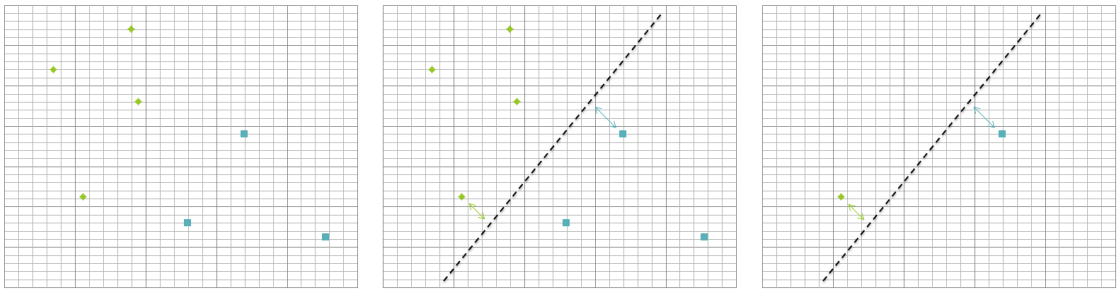

Figura 50. Classificadores de máxima margem. Na imagem da direita, pontos cujos multiplicadores de Lagrange são iguais a zero podem ser descartados da formulação final do classificador, e os pontos restantes podem ser determinados vetores de suporte.

de ser obtida considerando-se a distância entre os hiperplanos separadores de cada classe, ilustrados na Figura 50.

Assim, para maximizar a margem separadora, pode-se minimizar $\|\boldsymbol{\theta}\|^2$. No entanto, problemas reais são dificilmente linearmente separáveis. Isto leva a introdução de variáveis de folga ξ_i relacionadas com cada instância de treinamento a fim de se permitir erros nas restrições da margem. O problema então pode ser formulado como um problema de minimização restrita da forma

$$\min_{w,b,\xi} \frac{1}{2}\|\boldsymbol{\theta}\|^2 + C\left(\sum_{i=1}^{n} \xi_i\right)$$
$$\text{sujeito a } \begin{cases} y_i(\boldsymbol{\theta} \cdot \mathbf{x_i} + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \end{cases} \tag{5.11}$$

em que a constante C é um termo de regularização que impõe um peso a minimização dos erros no conjunto de treinamento em relação à minimização da complexidade do modelo. O termo C pode ser visto como o controle da capacidade da máquina de aprendizado. Introduzindo-se multiplicadores de Lagrange a_i, derivando em relação a $\boldsymbol{\theta}$, ξ e b, e impondo estacionaridade, pode-se obter o problema dual de otimização

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j}^{n} a_i a_j y_i y_j (\mathbf{x_i} \cdot \mathbf{x_j})$$
$$\text{sujeito a } \begin{cases} 0 \leq \alpha_i \leq C, \forall i = 1, \ldots, n \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases} \tag{5.12}$$

Assim, obtêm-se um problema de maximização convexo, expresso apenas a partir dos dados, incorporando controle da capacidade a partir da definição de C. Resolvendo este problema de maximização, os pontos x_i para os quais $\alpha_i > 0$ passam a serem denominados vetores de suporte e serão os únicos a fazer parte do classificador final. Pontos com $\alpha_i = 0$ poderão ser descartados, tornando o classificador final esparso. Assim, o classificador final em sua forma dual tem a forma esparsa

$$h(\mathbf{x}) = \text{sgn}\left(\sum_{\mathbf{x_j} \in SV} \alpha_i \, y_i \, \mathbf{x_j} \cdot \mathbf{x} + b\right) \tag{5.13}$$

em que SV denota o conjunto dos vetores de suporte identificados durante a otimização. Note que nesta discussão foi omitido o caso do termo independente b. Para obtê-lo, é necessário fazer uso das condições complementares de *Karush-Kuhn-Tucker*, cuja derivação de maneira muito mais completa é apresentada em (CRISTIANINI e SHAWE-TAYLOR, 2000).

5.2.4 Aplicação do truque do kernel

Em um de seus trabalhos, Cover (1965) notou que a probabilidade de se encontrar uma dicotomia linearmente separável aumenta com a dimensionalidade do espaço, o que pode ser notado considerando-se as equações

$$\begin{aligned} P(m,n) &= \frac{C(m,n)}{2^m} = \begin{cases} \frac{2}{2^m}\sum_{i=0}^{n}\binom{m-1}{i}, & m > n+1 \\ 1, & m \leq n+1 \end{cases} \\ C(m,n) &= \begin{cases} 2\sum_{i=0}^{n}\binom{m-1}{i}, & m > n+1 \\ 2^m, & m \leq n+1 \end{cases} \end{aligned} \tag{5.14}$$

Pode-se observar que, se $n \to \infty$, então $P \to 1$. Logo, fixando-se o número de pontos, quanto maior a dimensão, maior a probabilidade de se encontrar uma dicotomia linearmente separável. Assim, é natural consideramos uma transformação não linear $\varphi(\cdot): \mathbb{R}^n \to \mathcal{F}$ tal que, quando aplicada aos vetores de entradas $x_i \in \mathbb{R}^n$, os

projete a um espaço de atributos de alta dimensionalidade $\mathcal{F}$. Assim, o classificador apresentado na seção anterior pode tomar a forma

$$h(\mathbf{x}) = \operatorname{sgn}\left(\sum_{\mathbf{x_j} \in SV} \alpha_j \, y_j \, \langle \varphi(\mathbf{x_j}), \varphi(\mathbf{x_i}) \rangle \; + \; b\right). \tag{5.15}$$

Desta maneira, é possível notar que a função de decisão pode ser expressa utilizando-se apenas produtos internos. É interessante notar que não há necessidade de se calcular a transformação $\boldsymbol{\varphi}$ de maneira explícita. Todas as operações envolvendo produtos internos no espaço de alta dimensionalidade gerado por $\boldsymbol{\varphi}$ podem ser computados utilizando uma função *kernel* (MERCER, 1909) na forma

$$k(\mathbf{x}, \mathbf{z}) = \langle \varphi(\mathbf{x}), \varphi(\mathbf{z}) \rangle \tag{5.16}$$

em que esta função *kernel* representa um produto interno no espaço de atributos $\mathcal{F}$. Além do mais, como o mapeamento $\boldsymbol{\varphi(x)}$ não precisa mais ser computado, o espaço de atributos $\mathcal{F}$ pode ter dimensionalidade arbitrariamente alta, e inclusive ser infinito-dimensional (como no caso de um *kernel* Gaussiano). A Tabela 1 sumariza algumas escolhas comuns de funções *kernel*, com possíveis parametrizações e as respectivas descrições destes parâmetros.

Tabela 1. Exemplos de funções kernel

Função	**Nome**	**Parâmetros**
$k(\mathbf{x}, \mathbf{z}) = \mathbf{x} \cdot \mathbf{z}$	Linear	
$k(\mathbf{x}, \mathbf{z}) = (\mathbf{x} \cdot \mathbf{z} + c)^d$	Polinomial	Grau polinomial d, Constante de homogeneidade c
$k(\mathbf{x}, \mathbf{z}) = \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{x} - \mathbf{z}\|^2\right\}$	Gaussiano	Parâmetro de largura σ^2

É fácil verificar que, para a função *kernel* linear, pode-se retomar a forma original do classificador linear para um espaço Euclidiano. Muitas outras escolhas de funções kernel são apresentadas em (SOUZA, 2010), e uma descrição detalhada

de variedades e categorias de funções *kernel* está disponível em (GENTON, 2002). Sumarizando as operações realizadas anteriormente, pode-se, portanto, apresentar o classificador final na forma

$$\mathrm{h}(\mathbf{x}) = \mathrm{sgn}\left(\sum_{\mathbf{z_j} \in \mathrm{SV}} \alpha_\mathrm{j}\, \mathrm{y_j}\, \mathrm{k}(\mathbf{z_j}, \mathbf{x}) \;+\; \mathrm{b}\right). \tag{5.17}$$

Uma síntese dos passos envolvidos para se aplicar o truque do *kernel* é apresentada na Figura 51 a seguir.

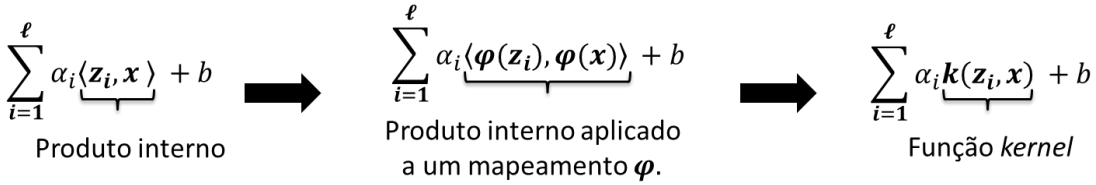

Figura 51. Diagrama em blocos exemplificando a aplicação do truque de kernel considerando-se ℓ vetores de suporte $\mathbf{z}_i$.

5.2.5 Aprendizado e estimação de parâmetros

Como apresentado na seção anterior, o problema de aprendizado em SVMs se resume a resolução de um problema de otimização quadrática cuja função objetivo a ser minimizada é também uma função convexa. Pode-se notar, portanto, que é possível utilizar qualquer otimizador quadrático para realizar o aprendizado das SVMs. Mas esta não é a abordagem mais eficiente possível. Métodos específicos para o treinamento de SVMs são capazes de explorar características particulares do problema de otimização apresentado e assim obterem soluções com menor esforço computacional. Dentre estes métodos está o método da Otimização Sequencial Mínima (*Sequential Minimal Optimization*, SMO), introduzido por Platt (1999) e posteriormente aprimorado por Keerthi, Shevade, *et al.* (2001). A abordagem adotada por Platt reduz o problema de otimização quadrática em uma bateria de pequenos problemas menores; de fato, os problemas tratados são mínimos por envolverem apenas dois multiplicadores de Lagrange a cada vez. Apesar de ter fornecido heurísticas a respeito da seleção dos multiplicadores a serem otimizados a cada iteração do algoritmo, Keerthi e colegas identificaram ineficiências em seu algoritmo origi-

nal. Para contornar estas ineficiências, estes últimos autores propuseram novas estratégias para a seleção dos pares a serem otimizados, como a seleção do par cuja violação das condições de otimalidade seja máxima a cada iteração.

Outro ponto de discussão surge acerca dos parâmetros C e da escolha da função kernel k. Em suma, podemos dizer que não há uma regra fácil para a obtenção destes parâmetros, e em geral o uso de validação cruzada (*Cross-Validation*, CV) juntamente de uma busca em grade (*Grid-Search*, GS) é recomendado para auxiliar sua escolha. Uma estratégia interessante para a obtenção de C foi proposta em (HASTIE, ROSSET, *et al.*, 2004). Em sua abordagem, estes autores derivaram um método para o cálculo do desempenho de todos os possíveis valores de C em um único passo, reduzindo grandemente o esforço computacional que seria desperdiçado em uma custosa busca em grade. Apesar de não fornecer uma escolha ótima, uma possível abordagem para encontrar valores razoáveis para a constante C pode ser obtida a partir de n amostras $\{x_i\}$ através da heurística $C^* = n/\sum_i^n k(x_i, x_i)$. Para o caso da escolha de parâmetros para uma função kernel Gaussiana, os autores Caputo, Sim, *et al.* (2002) propuseram uma heurística para a escolha do parâmetro σ^2 baseada no intervalo interquartil das estatísticas de norma dos dados de amostra.

5.2.6 O caso de múltiplas classes

Uma das maiores desvantagens do uso de SVMs está em sua aplicabilidade para problemas de múltiplas classes. Por serem inerentemente binárias, diversas abordagens e modificações a formulação original das SVMs foram propostas de maneira a torná-las aplicáveis nestas situações (VAPNIK, 1998; CRAMMER e SINGER, 2002). Parte desta dificuldade pode ser explicada devido ao fato da formulação original das SVMs não fornecer uma interpretação probabilística de suas saídas, mas sim apenas a distância do ponto classificado até sua margem de separação. Após a estabilização de sua formulação original, diversas outras abordagens surgiram; tais como o aprendizado de SVMs baseados em medidas de desempenho multivaloradas (JOACHIMS, 2005), probabilidades (PLATT, 1999; LIN, LIN e WENG, 2007; WU, LIN e WENG, 2003), e em particular, abordagens para sua generalização a problemas de múltiplas classes (PLATT, CRISTIANINI e SHAWE-TAYLOR, 2000; FRANC e HLAVAC, 2002).

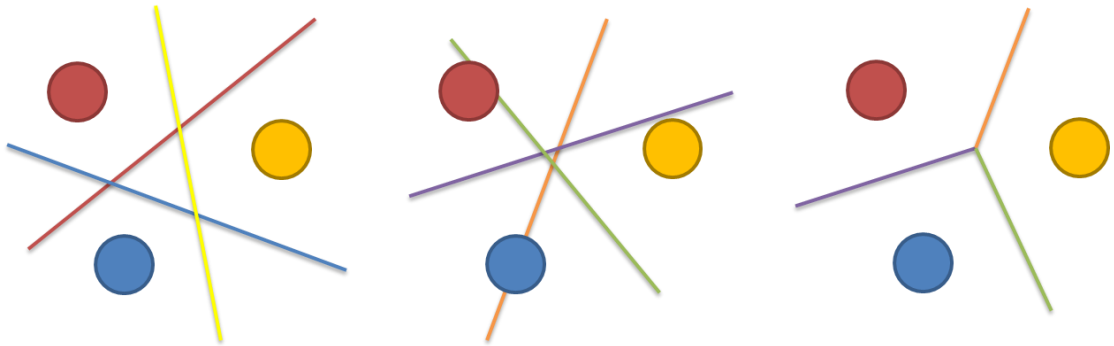

Figura 52. Abordagens de separação em múltiplas classes através do emprego de classificadores binários. A mistura das cores identifica a qual classificador binário pertence cada superfície de decisão. Da esquerda para direita: Abordagem um-contra-todos, abordagem par-a-par, abordagem por funções discriminantes.

Nesta seção, será possível notar que muito da teoria aplicada à generalização das SVMs a problemas de múltiplas classes nos remete a pesquisa de base inicial em reconhecimento de padrões, como os experimentos de Duda e Machanik (1963) e outras abordagens para tratamento de múltiplas classes como as citadas no estado da arte do final da década de 60 (NAGY, 1968). Pode-se até mesmo notar que muitas das técnicas têm sido redescobertas continuamente ao longo dos anos em diversas áreas do conhecimento, como por exemplo, no uso de funções discriminantes ou códigos de saída corretores de erros (*Error-Correcting Output Codes*, ECOC) (DIETTERICH e BAKIRI, 1995; FELTY e JURS, 1973).

Uma abordagem comum para o tratamento de múltiplas classes com SVMs, primeiro proposta por Vapnik (1995), é a abordagem *um-contra-todos* (*one-against-all*), representada, entre outras, na Figura 52. Nesta abordagem, um problema de decisão entre c classes é dividido em c subproblemas binários de decisão. Os dados são particionados de maneira que cada subproblema k apresente o rótulo da classe positiva (+1) para os dados pertencentes à classe ω_k e rótulos de classe negativa (-1) para todos os demais dados pertencentes as demais classes $\omega_{i \neq k}$. Um classificador binário $h_k(x)$ é treinado para cada um destes subproblemas.

No entanto, existem certas dificuldades ao se adotar estas abordagens. Em primeiro lugar, torna-se necessário o aprendizado de c classificadores no mesmo conjunto X de dados de treinamento, implicando em um aumento linear do tempo de processamento durante o *aprendizado e decisão*. Há ainda o problema já citado das SVMs não produzirem saídas probabilísticas, mas apenas informarem a distância do ponto até a margem de separação. É difícil visualizar que a tomada do maior valor $f_k(x)$ seja um bom indicador da classe caso a distribuição das classes não seja balanceada. Como argumentam Lee, Lin e Wahba (2004), mesmo que cada SVM

apresente a decisão ótima na discriminação de cada classe em particular do restante das classes, a seleção do valor máximo não implicaria que esta decisão continue sendo a decisão ótima para o problema original de separação em c classes.

Outra abordagem possível para o problema é a decomposição do problema original de c classes em $c(c-1)/2$ subproblemas menores. Esta abordagem é comumente conhecida como a estratégia *um-contra-um* (*one-against-one*) para problemas de múltiplas classes, e é representada pela figura central apresentada na Figura 52. Nesta abordagem, o problema original é decomposto em todos os possíveis problemas de classificação par-a-par. No entanto, explorando-se a simetria da matriz de decisão de todos os possíveis pares, podemos descartar decisões redundantes e desnecessárias e considerar apenas $c(c-1)/2$ problemas de decisão binária, conforme ilustrado na Figura 53.

Em comparação a abordagem *um-contra-todos*, em que é necessário aprender apenas c máquinas de decisão, nesta abordagem é necessário aprender $c(c-1)/2$ classificadores, o que a primeira vista parece tornar o método mais custoso e, portanto, menos atrativo. Mas o ponto crucial desta abordagem reside no fato destes $c(c-1)/2$ problemas serem significantemente *menores* do que os c problemas da abordagem *um-contra-todos* inicial. Como cada subproblema incorpora apenas dados de duas classes a cada vez, seu custo de aprendizado é reduzido. No entanto, como notado por Friedman (1996), sua variância aumenta devido a redução do número de amostras, ocasionando *overfitting* com maior facilidade. Esta dificuldade pode todavia ser contornada através de uma seleção criteriosa da capacidade de (dimensão VC) de cada classificador.

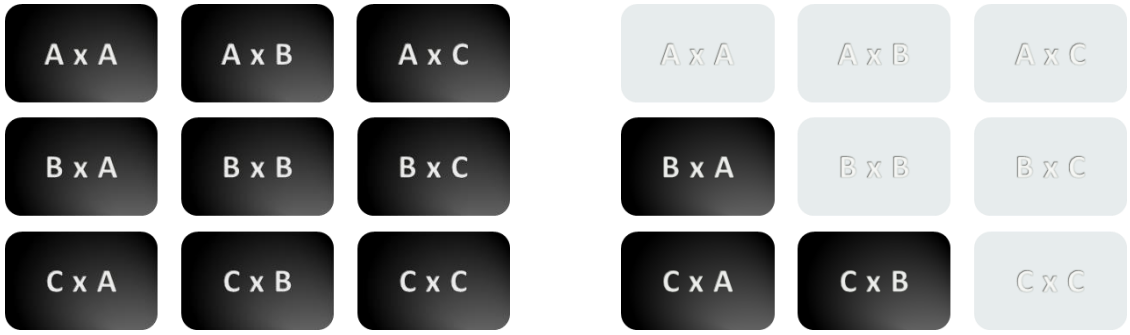

Figura 53. Decomposição de um problema original de três classes em todas suas possíveis combinações. Dada à simetria da decisão, apenas 3(3-1)/2 = 3 problemas precisam ser considerados.

Na abordagem *um-contra-um*, existem diversas maneiras de se combinar estes classificadores de maneira a se extrair uma única decisão. Uma das maneiras mais comuns é a regra de votação (FRIEDMAN, 1996). Na decisão por votação, todos os modelos são questionados de forma paralela, e a decisão de cada máquina a favor é utilizada para incrementar um painel de votação. A seleção com maior número de votos é então selecionada como melhor representante do rótulo verdadeiro da amostra apresentada. No entanto, ainda assim na etapa de decisão temos um número potencialmente alto de subproblemas a considerar, o que pode tornar o método pouco competitivo em comparação a métodos de tempo de interrogação constante como redes neurais, em que a complexidade da etapa de decisão depende da arquitetura escolhida para a rede.

Como bem observam Platt, Cristianini e Shawe-Taylor (2000), a complexidade do classificador *um-contra-um* pode crescer de maneira superlinear em relação ao tamanho de classes c e pode se tornar vagaroso durante a etapa de decisão de grandes problemas. De maneira à circunverter este problema, estes autores propuseram uma nova abordagem de classificação que é capaz de escalar linearmente ao tamanho de classes c explorando características de classificadores de máxima margem. Sua abordagem é baseada em Grafos Acíclicos Direcionados de Decisão (*Decision Acyclic Directional Graph*, DDAG). A classificação por DDAGs emprega uma estratégia de eliminação, em contraste com a estratégia de votação utilizada por Friedman. Na estratégia por eliminação, a cada problema de decisão, uma das classes é eliminada do processo de classificação. Esta abordagem é facilmente justificada, como exibido na Figura 54, quando cada problema de decisão é resolvido através de um classificador por hiperplanos. Deve-se notar que este é justamente o caso corrente quando empregamos classificadores de máxima margem como as SVMs.

Utilizando-se uma estrutura em grafo – como exibido na Figura 55 – para dispor cada um dos $(c-1)/2$ classificadores da formulação *um-contra-um*, pode-se obter um classificador para múltiplas classes em que seja possível obter a decisão final de uma classe em no máximo $c-1$ passos. É interessante notar que DDAGs generalizam o conceito de Árvores de Decisão (*Decision Tree*, DT), pois DDAGs permitem a presença de ciclos não direcionados. Desta maneira, é possível juntar o treinamento menos custoso da formulação *um-contra-um* ao mesmo tempo em que é possível explorar um reduzido tempo de computação durante a avaliação do modelo. Esta formulação é de particular interesse em aplicações que necessitem um rápido tempo de classificação, como em aplicações de tempo real.

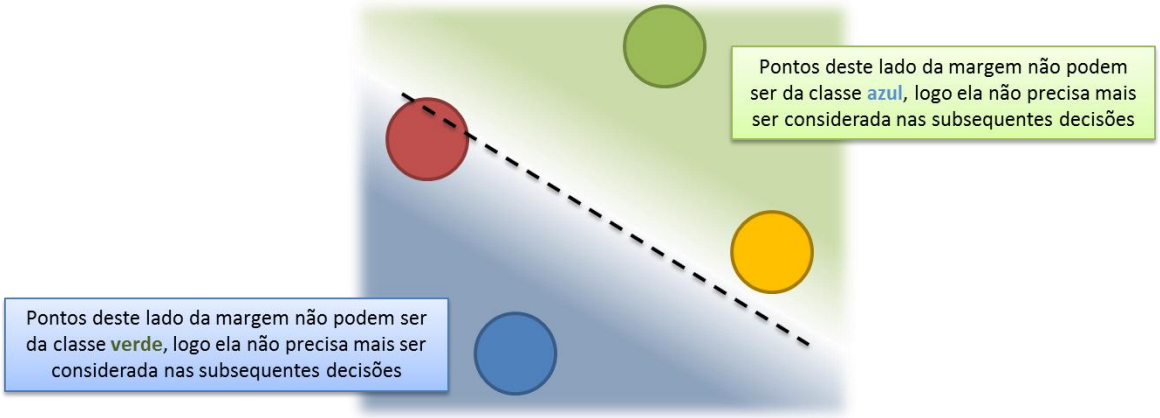

Figura 54. Decisão guiada por eliminação, imagem recriada a partir do trabalho original em (PLATT, CRISTIANINI e SHAWE-TAYLOR, 2000).

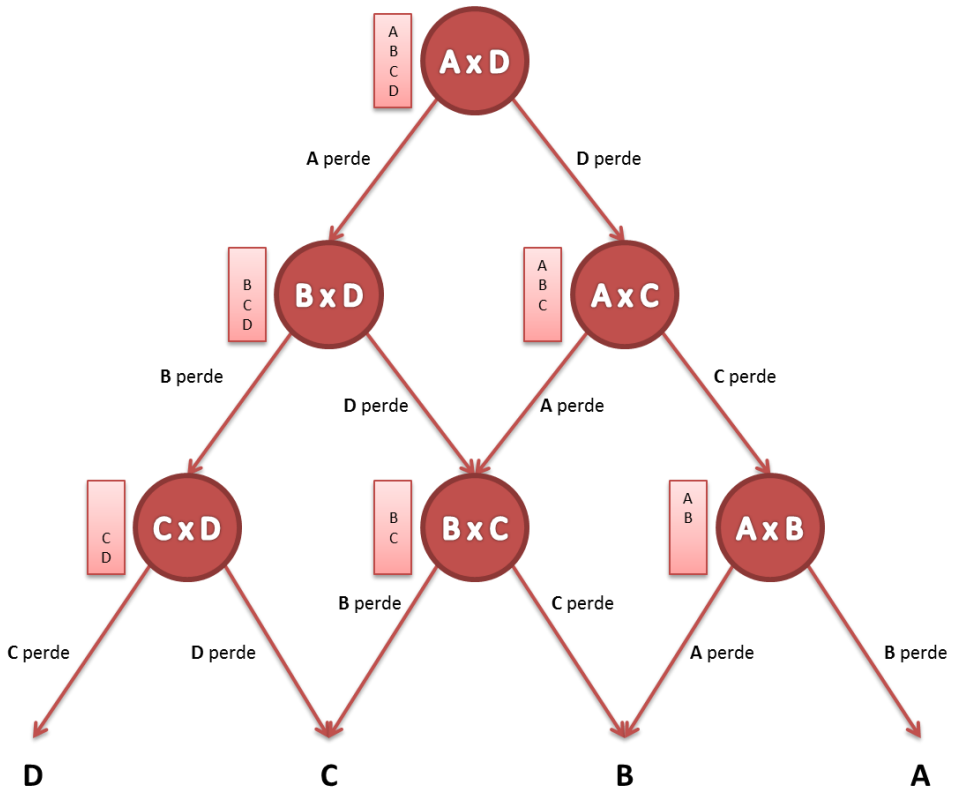

Figura 55. Decisão por grafos acíclicos direcionados, recriada a partir do trabalho original em (PLATT, CRISTIANINI e SHAWE-TAYLOR, 2000).

5.3 Modelos Ocultos de Markov

Modelos de Markov modelam sequências de observações ao decorrer do tempo. Um processo no tempo demonstra propriedade de Markov se a densidade de probabilidade condicional do evento corrente, dado todos os eventos passados e presentes, depender somente no i-ésimo evento mais recente (BISHOP, 2007, p. 607). Assumir a hipótese de que um evento no tempo depende apenas em um número limitado de eventos anteriores leva a uma estrutura matemática eficiente para expressar informações espaço-temporais. Esta estrutura permitiu, por exemplo, a descoberta de algoritmos bastante eficientes para estimação e avaliação destas probabilidades na forma dos algoritmos de *Baum-Welch* e de *Viterbi* (BAUM, PETRIE, *et al.*, 1970; VITERBI, 1967).

Os modelos ocultos de Markov são modelos duplamente estocásticos, no sentido em que modelam a probabilidade conjunta de duas variáveis aleatórias: a distribuição das observações $\boldsymbol{x} = \langle x_1, x_2, \cdots, x_T \rangle$ e a sua relação com o tempo na forma da sequência de estados ocultos $\boldsymbol{y} = \langle y_1, y_2, \cdots, y_T \rangle$, em que cada estado oculto y_t é pertencente a um conjunto finito de estados $\boldsymbol{S}$ como em

$$\mathrm{p}(\mathbf{x}, \mathbf{y}) = \prod_{\mathrm{t}=1}^{\mathrm{T}} \mathrm{p}(\mathrm{y_t}|\mathrm{y_{t-1}})\, \mathrm{p}(\mathrm{x_t}|\mathrm{y_t})\,. \tag{5.18}$$

Tal modelo assume que a sequência de estados $\boldsymbol{y}$ seja oculta e, portanto, não observável durante sua estimação. Isto significa que, para calcularmos a probabilidade $\boldsymbol{p(x)}$ de uma sequência de observações $\boldsymbol{x}$ é preciso, de alguma maneira, contornar o fato de que desconhecemos $\boldsymbol{y}$**.** Uma possível saída para contornar este problema é marginalizar $\boldsymbol{y}$ de maneira a se obter a probabilidade da sequência $\boldsymbol{x}$ considerando todas as sequências de estados possíveis.

$$\mathrm{p}(\mathbf{x}) = \sum_{\mathbf{y}} \mathrm{p}(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{y}} \prod_{\mathrm{t}=1}^{\mathrm{T}} \mathrm{p}(\mathrm{y_t}|\mathrm{y_{t-1}})\, \mathrm{p}(\mathrm{x_t}|\mathrm{y_t}) \tag{5.19}$$

A primeira vista, a consideração de todas as sequências de estados ocultos possíveis parece ser um problema intratável. No entanto, o cálculo desta probabilidade pode ser feito de forma elegante e em tempo linear ao tamanho T de $\boldsymbol{x}$ utili-

zando-se os algoritmos de *Forward* ou *Backward*, apresentados com bastante detalhes em (RABINER e JUANG, 1993) e (BISHOP, 2007).

Como mostra a equação (5.19), modelos ocultos de Markov operam utilizando distribuições de probabilidade para modelar a probabilidade de emissão de uma observação. No caso discreto, estas distribuições de probabilidade podem ser descritas como distribuições discretas de um conjunto finito de símbolos $\mathbf{\Lambda}$. Esta descrição pode ser representada pela quíntupla

$$\lambda = (\mathrm{n}, \mathrm{m}, \mathbf{A}, \mathbf{B}, \boldsymbol{\pi}), \tag{5.20}$$

em que

- n é o numero de estados para o modelo;
- m é o número de símbolos para observações distintas em cada estado;
- $\boldsymbol{A}$ é a distribuição de probabilidades de transição entre estados, dada na forma de uma matriz $n \times n$ com $\boldsymbol{A} = \{ a_{ij} \}$, em que cada a_{ij} representa a probabilidade de transição do estado i para o estado j, com $i, j \leq n$;
- $\boldsymbol{B}$ é a distribuição de probabilidades de emissão de cada símbolo do alfabeto; no caso discreto, é dada na forma de uma matriz $n \times m$ com $\boldsymbol{B} = \{ \boldsymbol{b_j}(o) \}$, em que cada vetor $\boldsymbol{b_j}$ contém a tabela de frequências modelando a emissão de cada símbolo $o \in \mathbf{\Lambda}$ para o estado j, com $j \leq n$;
- $\boldsymbol{\pi}$ é o vetor de probabilidades para o estado inicial, com $\boldsymbol{\pi} = \{ \boldsymbol{\pi}_i \}$, $i \leq n$.

Na parametrização apresentada acima, n indica a cardinalidade do conjunto de estados ocultos $\boldsymbol{S}$ e o mesmo pode ser dito de m a respeito do alfabeto finito $\mathbf{\Lambda}$. Entretanto, omitindo-se o número de estados n e o tamanho do alfabeto m, podemos expressar o conjunto de parâmetros de um HMM em uma forma compacta que pode servir para designar tanto modelos discretos como contínuos dependendo apenas na definição de $\boldsymbol{B}$, como simplesmente

$$\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi}). \tag{5.21}$$

No caso contínuo, $\boldsymbol{b_j}$ pode assumir qualquer distribuição de probabilidade, tal como uma Gaussiana ou mesmo uma mistura de Gaussianas de qualquer dimensão d. Neste caso, cada observação $\boldsymbol{x_t}$ de uma sequência $\boldsymbol{x}$ é assumida real e multivalorada tal que $\boldsymbol{x_t} \in \mathbb{R}^d$.

No entanto, uma desvantagem significativa dos modelos contínuos utilizando misturas de densidades é a aprendizagem de seus parâmetros, que se torna computacionalmente intensiva. Outra desvantagem é que o número de componentes de cada mistura deve ser determinado antes do aprendizado. Para reduzir o esforço computacional, podem-se utilizar hipóteses simplificadoras como a hipótese de independência entre o vetor de observações, o que nos leva a uma abordagem similar a do classificador de *Naïve Bayes* para modelar distribuições multidimensionais, porém passível de resultar em classificadores com menor poder de expressão. É importante mencionar que todos os parâmetros de um HMM devem obrigatoriamente formar probabilidades, o que complica ainda mais seu aprendizado.

5.3.1 Problemas canônicos

Existem três problemas canônicos associados aos modelos ocultos de Markov. A solução de cada um destes problemas revela a real utilidade e poder destes modelos. Estes problemas são:

- **Avaliação:** Dado um modelo λ, e uma sequência de observações $\boldsymbol{x} = \langle x_1, x_2, \cdots, x_T \rangle$, em que T é o tamanho desta sequência de observações, computar a probabilidade $p(\boldsymbol{x};\ \lambda)$ de ocorrência desta sequência.
- **Decodificação:** Dado um modelo λ e uma sequência de observações $\boldsymbol{x} = \langle x_1, x_2, \cdots, x_T \rangle$, encontrar a sequência de estados $\boldsymbol{y} = \langle y_1, y_2, \cdots, y_T \rangle$, $y \in S$ com maior chance de ter gerado $\boldsymbol{x}$.
- **Estimação:** Dada uma sequência de observações $\boldsymbol{x} = \langle x_1, x_2, \cdots, x_T \rangle$, ou um conjunto destas sequências $\mathbf{X} = \{ \boldsymbol{x}_i \}$, estimar os parâmetros que melhor aproximam um modelo λ para estas sequências.

No problema de avaliação, computar esta probabilidade requer a soma de todas as possíveis sequências de estados no modelo. No entanto, esta soma pode ser realizada de maneira bastante eficiente através do algoritmo *Forward*. Este algoritmo é um exemplo clássico do uso de programação dinâmica, aplicável para qualquer modelo gráfico em forma de cadeia, e é também uma forma especializada do algoritmo de *soma-produto* para modelos gráficos gerais.

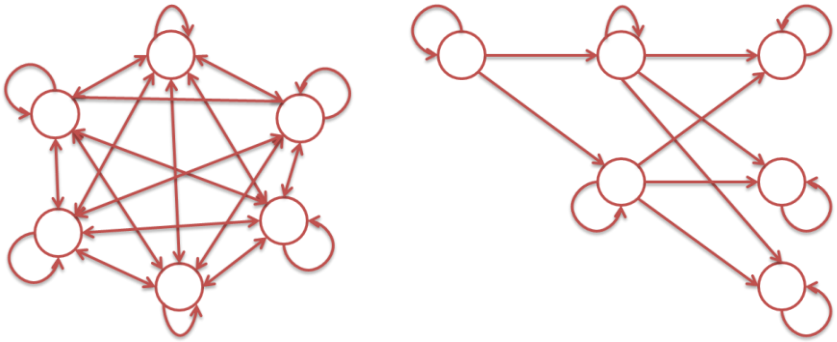

Figura 56. Diferentes topologias para um HMM. Esquerda: topologia ergódiga. Direita: topologia somente-a-frente (ou esquerdo-direita)

No problema de decodificação, encontrar a sequência requerida envolve encontrar um máximo dentre todas as possíveis sequências de estados. No entanto, novamente, este problema pode ser resolvido de maneira eficiente empregando-se o algoritmo de *Viterbi*, novamente um exemplo de programação dinâmica. O algoritmo de Viterbi também é generalizável para vários tipos de modelos gráficos, se tornando conhecido como algoritmo de *máxima-soma* (ou *máximo-produto*).

Já o problema de estimação de parâmetros pode ser visto como um problema de aprendizado em que desejamos obter a estimativa de máxima verossimilhança para um modelo dado um conjunto de dados. Uma maneira de se obter esta estimativa consiste em empregar uma versão especializada do algoritmo de Expectação-Maximização (*Expectation-Maximization*, EM) denominada algoritmo de *Baum-Welch*. No caso de aprendizado *online*, existe ainda a possibilidade de se utilizar o algoritmo *de Baldi-Chauvin* (BALDI e CHAUVIN, 1994) e outras variantes do algoritmo de *Baum-Welch*.

5.3.2 Uso em classificadores

Modelos ocultos de Markov modelam sequências de observações no tempo e são capazes de devolver a probabilidade de que os parâmetros específicos do modelo tenham gerado uma determinada sequência de observações. Em outras palavras, podemos dizer que cada modelo λ é capaz de devolver uma probabilidade $p(\boldsymbol{x}|\lambda)$ para uma determinada sequência de observações $\boldsymbol{x}$. Esta observação leva intuitivamente a formulação de um classificador de sequências gerativo, isto é, um classificador que incorpora um modelo de $p(\boldsymbol{x})$ para derivar uma probabilidade condicional a ser usada em um problema de classificação.

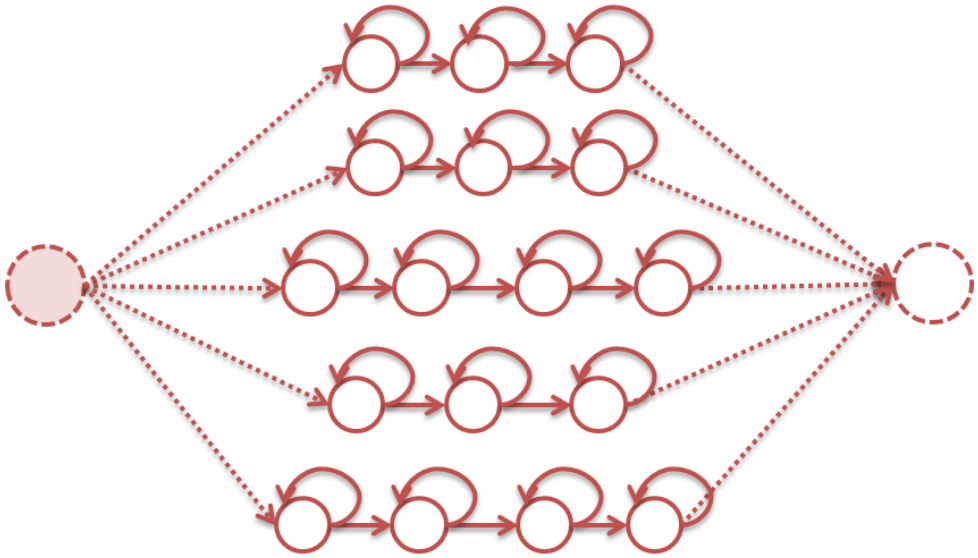

Figura 57. Representação da combinação de modelos de Markov através de uma representação em máquina de estados finitos.

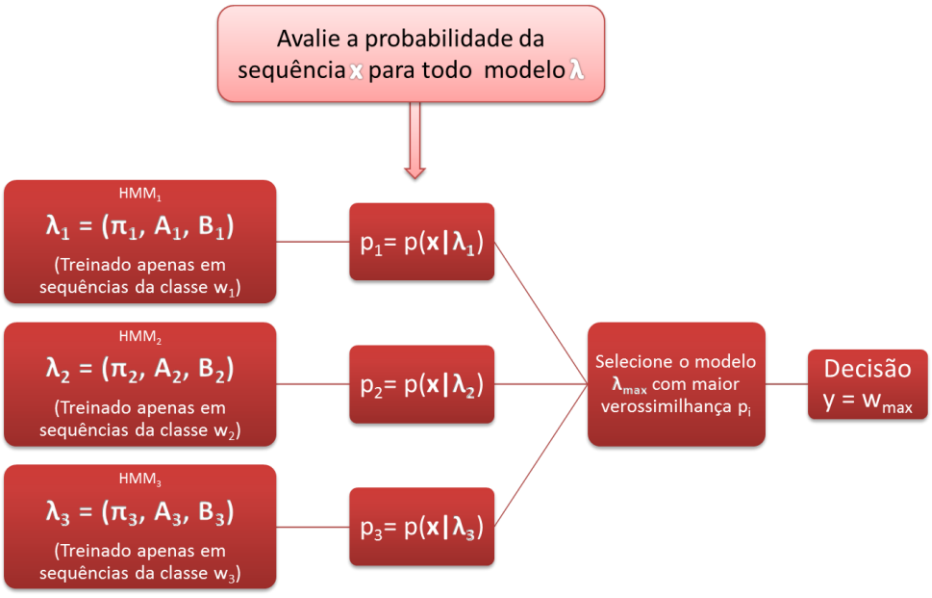

Figura 58. Representação em blocos da decisão de máxima verossimilhança (ML) baseada em múltiplos modelos de Markov, um para cada classe de sequências desejada.

Criando-se um modelo λ_i para cada classe de sequências $\omega_i \epsilon\, \Omega$, com $i \leq c$, podemos treinar cada um destes modelos para reconhecer cada uma das c classes possíveis. Tratando cada modelo λ_i como um modelo de uma densidade condicionada a classe ω_i, podemos formar a probabilidade condicional $p(\omega_i|x)$ para cada possível rótulo de classificação. É possível, então, utilizar um critério de máxima verossimilhança para obter a melhor estimativa $\hat{y}$ para a classe de uma nova sequência $\boldsymbol{x}$, conforme ilustrado na Figura 58.

No entanto, também podemos ir adiante e considerar alguma probabilidade *a priori* $p(\omega_i)$ para cada classe ω_i. Aplicando a regra de *Bayes,* obtemos

$$\mathrm{P}(\omega_\mathrm{i}|\mathrm{x}) = \frac{\mathrm{p}(\omega_\mathrm{i})\mathrm{p}(\mathrm{x}|\omega_\mathrm{i})}{\sum_\mathrm{j}^\mathrm{c} \mathrm{p}(\mathrm{x}|\omega_\mathrm{j})} \tag{5.22}$$

o que nos permite criar um modelo gerativo capaz de realizar discriminação utilizando-se o conhecimento *a priori* sobre a distribuição das classes $p(\omega_i)$ e um modelo oculto de *Markov* para cada uma das classes ω_i. A decisão MAP pode então ser computada através da maximização de argumento

$$\hat{\mathrm{y}} = \underset{\omega_\mathrm{j} \in\, \Omega}{\mathrm{argmax}}\, \mathrm{P}(\omega_\mathrm{j}|\mathbf{x}) \tag{5.23}$$

em que $P(\omega_j|\boldsymbol{x})$ é a probabilidade *a posteriori* de cada classe ω_j, Ω denota o conjunto de possíveis classes ω e $\hat{y}$ é a estimativa (predição) da classe verdadeira obtida pelo modelo. A classificação de sequências através de múltiplos modelos de Markov tem a vantagem na qual a inclusão de uma nova classe de sequências não requererá um novo treinamento de todos os modelos formando o classificador, mas sim apenas do novo modelo λ_{c+1} sendo adicionado.

5.4 Campos Aleatórios Condicionais

Modelos ocultos de Markov tentam modelar a distribuição conjunta $p(\boldsymbol{x}, \boldsymbol{y})$ das observações $\boldsymbol{x}$ e a sua relação com o tempo na forma da sequência de estados ocultos $\boldsymbol{y}$. Estes modelos podem ser combinados de maneira a formarem um classificador utilizando-se algum critério de decisão como o critério MAP ou ML. Porém conforme apresentado na seção 5.2, a estimação de densidades é um problema tipicamente mal condicionado e em prática muito difícil de resolver quando considera-

mos um número finito de amostras. É possível verificar que, para a simples tarefa de classificação, a estimação de $p(x)$ não se faz necessária, e seguindo o Princípio Primordial da SRM, se há apenas um conjunto de informação restrito para a resolução de algum problema, deve-se tentar resolver o problema diretamente e nunca resolver um problema mais geral como um passo intermediário. Veremos nesta seção como os Campos Aleatórios Condicionais (*Conditional Random Fields*, CRF), originalmente criados por Lafferty, McCallum e Pereira (2001) objetivam resolver o problema de modelar diretamente a probabilidade condicional $p(x|y)$, que é suficiente para se resolver o problema de classificação. Pode-se notar que há um crescente interesse na literatura por aplicação de CRFs, como os trabalhos conduzidos por Taskar, Abbeel e Koller (2002), Peng, Feng e McCallum (2004) e Peng e McCallum (2004), incluindo também as aplicações em visão computacional por Quattoni, Collins e Darrell (2005).

Apesar de existirem discordâncias quanto à aplicabilidade e desempenho de modelos gerativos contra modelos discriminativos (NG e JORDAN, 2001; XUE e TITTERINGTON, 2008), pode-se destacar que a mudança de paradigma de modelos gerativos para discriminativos no contexto do reconhecimento de fala desempenhou papel fundamental no avanço desta área, como bem evidenciam Juang e Rabiner (2005). Segundo estes autores[14], esta mudança foi motivada pelo reconhecimento do fato de que as distribuições de probabilidade que regem os sinais acústicos da fala não poderiam ser modeladas de maneira acurada, tornando a teoria de decisão de Bayes "inaplicável sob estas circunstâncias" (JUANG e RABINER, 2005, p. 16).

Nesta seção, será interessante primeiro inserir os modelos ocultos de Markov no contexto de modelos gráficos e apresentar as diferenças de modelos gerativos e discriminativos em termos de modelos gráficos direcionados e não direcionados. Seguindo a definição em (BISHOP, 2007, p. 360), modelos gráficos são modelos que tentam representar a interdependência entre variáveis aleatórias a partir de uma representação em grafos. Estes grafos podem ser direcionados ou não direcionados, sendo que a direção de cada aresta representa alguma forma de causalidade entre as variáveis. O objetivo geral de um modelo gráfico, ainda segundo Bishop, é capturar esta maneira com a qual a probabilidade conjunta destas variáveis aleatórias pode ser decomposta em um produto de fatores que dependem somente de subconjunto

[14] L. R. Rabiner foi um dos pioneiros na criação e uso dos modelos ocultos de Markov, sendo o primeiro a publicar o algoritmo de *scaling* para computação do método de *Forward-Backward* em seu altamente citado *"tutorial on hidden Markov models"* (RABINER, 1990).

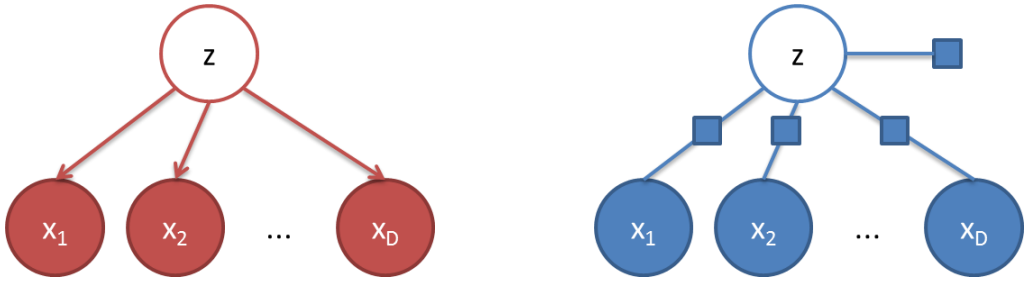

Figura 59. Esquerda: modelo gráfico do classificador Naïve Bayes como um modelo gráfico direcionado. Direita: modelo gráfico para o classificador Naïve Bayes como um modelo gráfico não direcionado, também denominado modelo de Regressão Logística.

destas variáveis. Desta maneira, torna-se possível obter uma compreensão simplificada da interação entre as variáveis sem ser necessário considerar o grafo completo, como é o caso da cobertura de Markov para grafos direcionados acíclicos.

Como citado anteriormente, modelos gráficos podem ser direcionados ou não direcionados. Modelos gráficos direcionados, quando acíclicos, são também denotados Redes Bayesianas (*Bayesian Networks*), cujo exemplo mais famoso é sem dúvida o classificador de Naïve Bayes. Já modelos gráficos não direcionados são também conhecidos como Campos Aleatórios Markovianos (*Markov Random Fields*, MRF), conforme apresentados em (KINDERMANN e SNELL, 1980). Um dos modelos não direcionados mais conhecidos é o modelo de regressão logística, muito popular em aplicações de estatística.

Utilizando-se a terminologia de (NG e JORDAN, 2001), pode-se notar que o modelo de regressão logística e o modelo de Naïve Bayes formam um par gerativo-discriminativo, no sentido que ambos classificadores consideram o mesmo espaço de hipóteses, e todo Naïve Bayes pode ser convertido em uma Regressão Logística com a mesma superfície de separação e vice-versa. A única diferença entre estes dois classificadores se dá na abordagem gerativa de um e discriminativa do outro (SUTTON e MCCALLUM, 2007). Os modelos de *Naïve Bayes* e de Regressão Logística são ilustrados na Figura 59.

Assim como o modelo de *Naïve Bayes*, os modelos ocultos de Markov também são considerados modelos gerativos e também exemplos de modelos gráficos direcionados. Seu par gerativo-discriminativo é formado em conjunto com CRFs de cadeia linear, conforme ilustrado na Figura 60 e discutido no decorrer desta seção.

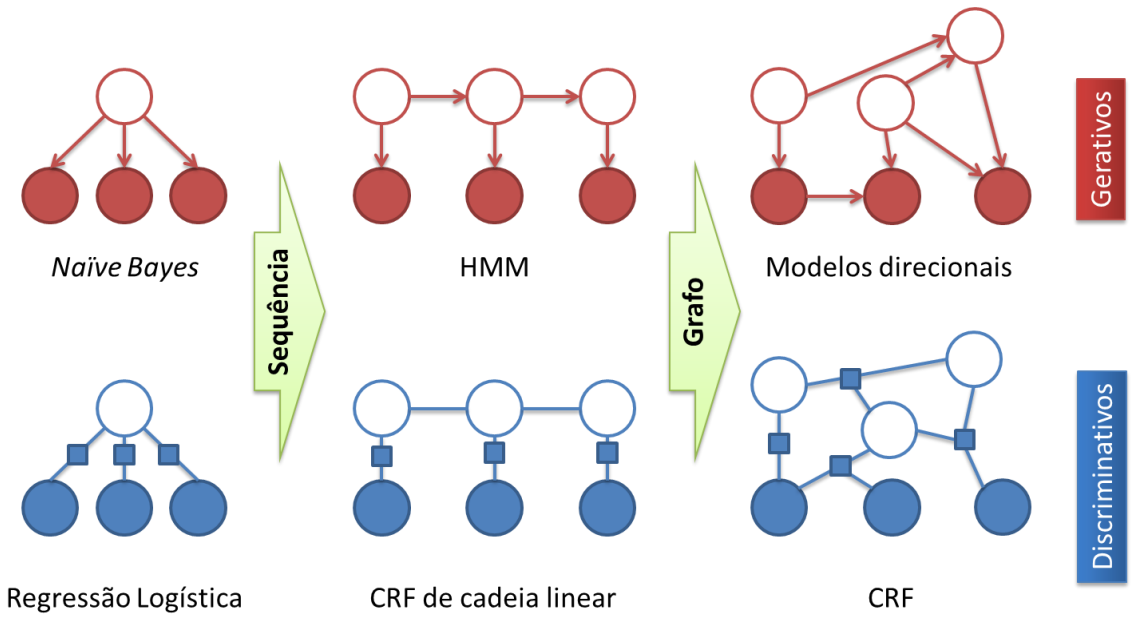

Figura 60. Diagrama das relações entre Naïve Bayes, regressão logística, HMMs, CRFs de cadeia linear, modelos gerativos direcionais e CRFs. Imagem recriada e adaptada a partir do trabalho original de (SUTTON e MCCALLUM, 2007).

Para compreender a formulação de CRFs, podemos primeiro adotar uma definição de MRFs baseada na definição apresentada[15] em (SUTTON e MCCALLUM, 2007).

Seja $\boldsymbol{V}$ um conjunto de variáveis aleatórias, formado por dois subconjuntos distintos, $\boldsymbol{X}$ e $\boldsymbol{Y}$, de maneira tal que $\boldsymbol{V} = \boldsymbol{X} \cup \boldsymbol{Y}$. As variáveis aleatórias contidas no conjunto $\boldsymbol{X}$ serão consideradas variáveis de entrada, independentes, assumidas observáveis. As variáveis aleatórias contidas em $\boldsymbol{Y}$ serão consideradas variáveis de saída, assumidas dependentes das variáveis $\boldsymbol{X}$, no sentido que seja possível extrair alguma relação $h: \boldsymbol{X} \rightarrow \boldsymbol{Y}$.

Consideraremos que toda variável aleatória $\boldsymbol{v} \in \boldsymbol{V}$ possa assumir valores de um conjunto $\boldsymbol{\mathcal{V}_v}$, que tanto pode ser contínuo como discreto, e que uma atribuição em particular para $\boldsymbol{X}$ será denotada por $\boldsymbol{x}$. Isto significa que, para cada i-ésimo elemento x_i de uma atribuição $\boldsymbol{x}$, haverá um conjunto associado e distinto $\boldsymbol{\mathcal{V}_i}$ de possíveis valores a que este elemento está restrito. O produto cartesiano de todos os conjuntos $\boldsymbol{\mathcal{V}_v}$ será denotado $\mathbb{V}$, de maneira que $\mathbb{V} = \prod_v \boldsymbol{\mathcal{V}_v}$.

[15] Ao longo destas seções, a maioria das definições será baseada no compreensivo material publicado por (SUTTON e MCCALLUM, 2007).

Consideraremos também que uma atribuição $\boldsymbol{x}$ pertencente a um subconjunto $\boldsymbol{C} \subset \boldsymbol{X}$ será denotada por $\boldsymbol{x_c}$, e que estas mesmas definições se apliquem às variáveis em $\boldsymbol{Y}$. Dada uma coleção de subconjuntos $\boldsymbol{C} \subset \boldsymbol{V}$, torna-se então possível definir *modelos gráficos não direcionados* como o conjunto de todas as distribuições passíveis de serem escritas na forma

$$\mathrm{p}(\mathbf{x}, \mathbf{y}) = \frac{1}{\mathrm{Z}} \prod_{\mathrm{c}} \Psi_{\mathrm{c}}(\mathbf{x_c}, \mathbf{y_c}). \tag{5.24}$$

para qualquer escolha de fatores $F = \{\Psi_c\}$. Neste modelo, F denota o conjunto de *funções potenciais*, ou *fatores,* tal que $\Psi_c: \mathbb{V} \rightarrow \mathbb{R}^+$. Será assumido que as funções potenciais Ψ_c tenham a forma

$$\Psi_{\mathrm{c}}(\mathbf{x_c}, \mathbf{y_c}) = \exp\left\{\sum_{\mathrm{k}} \theta_{\mathrm{Ck}} \mathrm{f}_{\mathrm{Ck}}(\mathbf{x_c}, \mathbf{y_c})\right\} \tag{5.25}$$

em que $\boldsymbol{\theta_C} \in \mathbb{R}^k$ é um vetor de parâmetros e $\{f_{Ck}\}$ um conjunto de *funções características* $f: \mathbb{V} \rightarrow \mathbb{R}$ associadas ao subconjunto $\boldsymbol{C}$. O conjunto de funções de característica também pode ser denotado como um vetor de *estatísticas suficientes* definidas sobre $\boldsymbol{x_c}$ e $\boldsymbol{y_c}$. A constante Z presente na equação (5.24) pode ser vista um fator de normalização cujo objetivo é garantir que a distribuição esteja corretamente normalizada no intervalo unitário, sendo definida por

$$\mathrm{Z} = \sum_{\mathbf{x},\mathbf{y}} \prod_{\mathrm{c}} \Psi_{\mathrm{C}}(\mathbf{x_c}, \mathbf{y_c}) \,. \tag{5.26}$$

Nota-se que, no caso geral, a constante de normalização Z é muito difícil e até mesmo impraticável de ser calculada.

Esta definição para modelos gráficos não direcionados pode ser vista como uma generalização simples da definição comum de MRFs em que não é assumido nenhum particionamento das variáveis aleatórias $\boldsymbol{V}$ em variáveis dependentes $\boldsymbol{Y}$ e independentes $\boldsymbol{X}$. No entanto, esta distinção serve de base para que se considere uma versão discriminativa de MRFs em que seja possível modelar diretamente a relação $h: \boldsymbol{X} \rightarrow \boldsymbol{Y}$. Ao invés de modelarmos a probabilidade conjunta $\boldsymbol{p(x, y)}$, pode-se modelar diretamente a probabilidade condicional $\boldsymbol{p(y|x)}$ através da marginalização de $\boldsymbol{y}$, o que acaba resultando em um relaxamento da hipótese de Markov para

que $\boldsymbol{Y}$ possa ser apenas *condicionalmente* Markov dado $\boldsymbol{X}$. A definição a seguir foi adaptada de (LAFFERTY, MCCALLUM e PEREIRA, 2001) e provê uma definição mais concisa das ideias apresentadas.

> ***Definição.*** *Seja* $G = (V, E)$ *um grafo tal que* $\boldsymbol{Y} = (\boldsymbol{Y}_v)$, $v \in \boldsymbol{\mathcal{V}}$, *de maneira que* $\boldsymbol{Y}$ *seja indexado pelos vértices de* G. *Então* $(\boldsymbol{X}, \boldsymbol{Y})$ *é um campo aleatório condicional no caso em que, quando condicionado em* $\boldsymbol{X}$, *as variáveis aleatórias* $\boldsymbol{Y}_v$ *obedeçam à propriedade de Markov a respeito do grafo:* $p(\boldsymbol{Y}_v|\boldsymbol{X}, \boldsymbol{Y}_w,\ w \neq v) = p(\boldsymbol{Y}_v|\boldsymbol{X}, \boldsymbol{Y}_w,\ w \sim v)$, *em que* $w \sim v$ *significa que* w *e* v *são vizinhos em* G.

A partir desta definição, pode-se observar que um CRF especifica não um, mas sim uma família de MRFs, cada um dos quais associados a uma observação $\boldsymbol{x} \in \boldsymbol{X}$. Considerando a equação (5.24) e marginalizando em relação a $\boldsymbol{y}$, pode-se obter a formulação de CRFs na forma

$$\mathrm{p}(\mathbf{y}|\mathbf{x}) = \frac{\mathrm{p}(\mathbf{x}, \mathbf{y})}{\sum_{\mathbf{y}'} \mathrm{p}(\mathbf{x}, \mathbf{y}')} = \frac{\prod_{\mathrm{c}} \Psi_{\mathrm{c}}(\mathbf{x}_{\mathbf{c}}, \mathbf{y}_{\mathbf{c}})}{\sum_{\mathbf{y}'} \prod_{\mathrm{c}} \Psi_{\mathrm{C}}(\mathbf{x}_{\mathbf{c}}, \mathbf{y}_{\mathbf{c}})}, \tag{5.27}$$

e finalmente, tomando-se $Z(\boldsymbol{x}) = \sum_{\boldsymbol{y}'} \prod_{c} \Psi_c(\boldsymbol{x}_c, \boldsymbol{y}_c)$, pode-se apresentar uma formulação para CRFs em sua forma mais comum dada por

$$\mathrm{p}(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathrm{Z}(\mathbf{x})} \prod_{\mathrm{c}} \Psi_{\mathrm{c}}(\mathbf{x}_{\mathbf{c}}, \mathbf{y}_{\mathbf{c}}). \tag{5.28}$$

Uma definição ainda mais geral para a modelagem de CRFs também é apresentada em (SUTTON e MCCALLUM, 2007). Nesta nova definição, os fatores de G são particionados em $\mathcal{C} = \{C_1, C_2, \ldots, C_P\}$, em que cada C_p constitui um clique molde *(clique template)* cujos parâmetros das funções potenciais estão amarrados. Cada um destes cliques molde C_p especifica um conjunto de fatores contendo um conjunto de estatísticas suficientes $\{f_{pk}(\boldsymbol{x}_p, \boldsymbol{y}_p)\}$ e parâmetros $\boldsymbol{\theta}_p \in \Re^{K(p)}$, de maneira que o modelo geral de um CRF possa então ser escrito na forma

$$\mathrm{p}(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathrm{Z}(\mathbf{x})} \prod_{\mathrm{C_p} \in \mathcal{C}} \prod_{\Psi_{\mathrm{c}} \in \mathrm{C_p}} \Psi_{\mathrm{c}}(\mathbf{x}_{\mathbf{c}}, \mathbf{y}_{\mathbf{c}}; \theta_{\mathrm{p}}) \tag{5.29}$$

em que a função de partição $Z(\boldsymbol{x})$ é dada por

$$\mathrm{Z}(\mathbf{x}) = \sum_{\mathbf{y}} \prod_{\mathrm{C_p} \in \mathcal{C}} \prod_{\Psi_\mathrm{c} \in \mathrm{C_p}} \Psi_\mathrm{c}(\mathbf{x_c}, \mathbf{y_c}; \theta_\mathrm{p}) \tag{5.30}$$

e cada fator é parametrizado como

$$\Psi_\mathrm{c}(\mathbf{x_c}, \mathbf{y_c}; \theta_\mathrm{p}) = \exp\left\{\sum_{\mathrm{k}=1}^{\mathrm{K(p)}} \theta_{\mathrm{pk}} \mathrm{f_{pk}}\,(\mathbf{x_c}, \mathbf{y_c})\right\} \tag{5.31}$$

De maneira análoga ao problema de decodificação associado aos HMMs, o modelo descrito por um CRF é capaz de atribuir um rótulo y_t a cada observação $x_t \in \boldsymbol{x}$. De fato, podemos observar que a distribuição condicional originada pela marginalização de todas possíveis sequências de estados em um HMM prescreve justamente a especificação de um CRF. Pode-se demonstrar este fato a partir da escolha específica de um conjunto de funções características. Seguindo a demonstração de (SUTTON e MCCALLUM, 2007), o primeiro passo é reescrever (5.18) utilizando um conjunto de funções indicadoras $1_{\{x=x\prime\}}$, em que $1_{\{x=x\prime\}}$ retorna o valor 1 quando $\boldsymbol{x} = \boldsymbol{x'}$ e 0 no caso contrário, conforme demonstrado a seguir.

$$\mathrm{p}(\mathbf{x}, \mathbf{y}) = \frac{1}{\mathrm{Z}} \exp\left\{\sum_{\mathrm{t}}^{\mathrm{T}} \sum_{\mathrm{i,j}}^{\mathrm{N}} \lambda_{\mathrm{ij}} 1_{\{\mathrm{y_t}=\mathrm{i}\}} 1_{\{\mathrm{y_{t-1}}=\mathrm{j}\}} + \sum_{\mathrm{t}}^{\mathrm{T}} \sum_{\mathrm{i}}^{\mathrm{N}} \sum_{\mathrm{o}}^{\mathrm{M}} \mu_{\mathrm{oi}} 1_{\{\mathrm{y_t}=\mathrm{i}\}} 1_{\{\mathrm{x_t}=\mathrm{o}\}}\right\} \tag{5.32}$$

Escolhendo-se parâmetros $\boldsymbol{\theta} = \{\lambda, \mu\}$, tal que $\boldsymbol{\lambda}_{ij} = \log \boldsymbol{A}_{ij}$ e $\boldsymbol{\mu}_{oi} = \log \boldsymbol{B}_{ik}$, em que $\mathbf{A}$ e $\mathbf{B}$ são as matrizes de transição e de emissão discutidas na seção 5.3, pode-se notar que é possível expressar qualquer HMM utilizando esta formulação. Pode-se agora escolher $\{f_k\}$ tal que exista uma função característica $f_{ij}(y, y', x) = 1_{\{y=i\}} 1_{\{y\prime=j\}}$ para cada transição de estados (i, j) e uma função característica $f_{io}(y, y', x) = 1_{\{y=i\}} 1_{\{x=o\}}$ para cada par estado-símbolo de observação (i, o) de maneira que o HMM possa finalmente ser expresso na forma

$$\mathrm{p}(\mathbf{x}, \mathbf{y}) = \frac{1}{\mathrm{Z}} \exp\left\{\sum_{\mathrm{k}=1}^{\mathrm{K}} \lambda_\mathrm{k} \mathrm{f_k}(\mathrm{y_t}, \mathrm{y_{t-1}}, \mathrm{x_t})\right\} \tag{5.33}$$

que reflete precisamente a notação de um MRF apresentada em (5.24) e (5.25), podendo-se notar que (5.33) define exatamente a mesma família de distribuições que (5.18). Note que se marginalizarmos $\boldsymbol{y}$ para fora da distribuição conjunta, obtemos

$$\mathrm{p}(\mathbf{x}|\mathbf{y}) = \frac{\mathrm{p}(\mathbf{x},\mathbf{y})}{\sum_{\mathbf{y}'} \mathrm{p}(\mathbf{x},\mathbf{y}')} = \frac{\exp\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\}}{\sum_{\mathbf{y}'} \exp\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\}}, \tag{5.34}$$

que por sua vez é justamente a formulação de um CRF de cadeia linear, formando um par discriminativo-gerativo com os modelos ocultos de Markov apresentados na seção 5.3, como evidenciado em sua forma final

$$\mathrm{p}(\mathbf{x}|\mathbf{y}) = \frac{1}{\mathrm{Z}(\mathbf{x})} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\right\}. \tag{5.35}$$

Finalmente, podemos notar que, para modelos de cadeia linear, a constante de normalização $\boldsymbol{Z(x)}$, que considera todas as possíveis sequências de estados ocultos $\boldsymbol{y}$, pode ser computada de maneira eficiente utilizando-se os mesmos algoritmos de *Forward* e *Backward* usados por HMMs. Para modelos baseados em árvores, estas quantidades podem ser computadas através do algoritmo de Propagação de Crença (*Belief Propagation*) (PEARL, 1982). No caso de estruturas mais gerais, é necessário lançar mão de técnicas de aproximação como a Propagação de Crença Cíclica (*Loopy Belief Propagation*) (MURPHY, WEISS e JORDAN, 1999).

5.4.1 Aprendizado

Na última seção levantamos o fato de que modelos gerativos podem ser convertidos em modelos discriminativos quando estes formam um par gerativo-discriminativo. Desta maneira, podemos sempre converter um HMM já treinado em um CRF de cadeia linear. No entanto, nem sempre podemos converter CRFs de cadeia linear em HMMs, pois seus parâmetros nem sempre formam as probabilidades requeridas pelos modelos gerativos. Pode-se dizer que HMMs especificam apenas alguns dos modelos passíveis de serem especificados por CRFs, o que seria esperado, por exemplo, ao notar-se que CRFs podem apresentar covariâncias negativas no caso de vetores de característica contendo momentos de segunda ordem, ou mesmo matrizes de covariância não positivo-definidas no caso de vetores de caracte-

rísticas multidimensionais. De qualquer modo, vemos que é possível inicializar um CRF de cadeia linear utilizando-se um HMM, e nesta seção veremos como proceder para realizar seu treinamento como um problema de otimização irrestrita.

Para realizar o treinamento de HMMs são necessárias técnicas que respeitem a estrutura probabilística de seus parâmetros, de maneira que a distribuição conjunta modelada por estes modelos continue válida. No caso de CRFs, podemos maximizar diretamente uma função objetivo, o que nos permite utilizar qualquer técnica de otimização numérica para realizar seu treinamento. Nesta seção, assumiremos que os CRFs possuam uma estrutura em cadeia linear, novamente seguindo muito do trabalho de Sutton e McCallum (2007).

Supondo um problema de aprendizado supervisionado, consideraremos um conjunto *i.i.d* de N instâncias de treinamento na forma

$$\mathcal{D} = \{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}, 1 \leq i \leq N \tag{5.36}$$

em que cada $\boldsymbol{x}^{(i)}$ seja um vetor de observações $\boldsymbol{x}^{(i)} = \langle x_1^{(i)}, x_2^{(i)}, \ldots, x_T^{(i)} \rangle$ e cada $\boldsymbol{y}^{(i)}$ seja o vetor de rótulos $\boldsymbol{y}^{(i)} = \langle x_1^{(i)}, x_2^{(i)}, \ldots, x_T^{(i)} \rangle$, com cada estado $y_t^{(i)}$ de $\boldsymbol{y}^{(i)}$ associado a sua respectiva observação $x_t^{(i)}$ de $\boldsymbol{x}^{(i)}$. Como CRFs modelam probabilidades condicionais, uma escolha adequada de função objetivo pode ser dada pela maximização da verossimilhança condicional dos dados, expressa por

$$\ell(\theta) = \sum_{i=1}^{N} \log p(\mathbf{y}|\mathbf{x}) \tag{5.37}$$

Para o caso de cadeia linear, $\boldsymbol{\ell(\theta)}$ pode ser expressa por

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{N} \sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k (y_t^{(i)}, y_{t-1}^{(i)}, x_t^{(i)}) - \sum_{i=1}^{N} \log Z(\mathbf{x}^{(i)}) \tag{5.38}$$

Para incorporar controle da capacidade, podemos adicionar um termo de regularização parametrizado de acordo com σ^2, o que permite uma interpretação Bayesiana como a definição de uma distribuição *a priori* sobre seus parâmetros, levando à expressão

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{k=1}^{K}\lambda_k f_k\left(y_t^{(i)}, y_{t-1}^{(i)}, x_t^{(i)}\right) - \sum_{i=1}^{n}\log Z\left(\mathbf{x}^{(i)}\right) - \sum_{k=1}^{K}\frac{\lambda_k^2}{2\sigma^2}. \tag{5.39}$$

Derivando $\ell(\boldsymbol{\theta})$a respeito de todos os parâmetros $\boldsymbol{\lambda} \in \boldsymbol{\theta}$, obtemos o vetor gradiente $\boldsymbol{g} = \langle\frac{\partial\ell}{\partial\lambda_1}, \frac{\partial\ell}{\partial\lambda_2}, \ldots, \frac{\partial\ell}{\partial\lambda_K}\rangle$, $1 \leq k \leq K$, em que cada derivada parcial $\frac{\partial\ell}{\partial\lambda_k}$ é dada por

$$\frac{\partial\ell}{\partial\lambda_k} = \sum_{i=1}^{N}\sum_{t=1}^{T}\lambda_k f_k\left(y_t^{(i)}, y_{t-1}^{(i)}, x_t^{(i)}\right) - \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{y,y'} f_k\left(y, y', x_t^{(i)}\right) p\left(y, y' | x_t^{(i)}\right) - \frac{\lambda_k}{\sigma^2}. \tag{5.40}$$

No caso de árvores ou estruturas gráficas mais gerais, a função de verossimilhança condicional é expressa por

$$\ell(\boldsymbol{\theta}) = \sum_{C_p \in \mathcal{C}}\sum_{\Psi_c \in C_p}\sum_{k=1}^{K(p)}\lambda_{pk} f_{pk}\left(\mathbf{x_c}, \mathbf{y_c}\right) - \log Z(\mathbf{x}) \tag{5.41}$$

e as derivadas parciais formando seu vetor gradiente são dadas por

$$\frac{\partial\ell}{\partial\lambda_{pk}} = \sum_{\Psi_c \in C_p} f_{pk}(\mathbf{x_c}, \mathbf{y_c}) - \sum_{\Psi_c \in C_p}\sum_{\mathbf{y'}_c} f_{pk}\left(\mathbf{x_c}, \mathbf{y'}_c\right) p\left(\mathbf{y'}_c | \mathbf{x}\right). \tag{5.42}$$

Em ambos os casos, a função objetivo definida por $\ell(\boldsymbol{\theta})$ é côncava, e, como tal, resulta em um problema de otimização convexo em que qualquer ótimo local é também um ótimo global.

5.5 Campos Aleatórios Condicionais Ocultos

Na seção anterior detalhamos os aspectos e o funcionamento de CRFs e vimos como HMMs e CRFs de cadeia linear formam um par gerativo-discriminativo. Nesta seção, veremos que Campos Aleatórios Condicionais Ocultos (*Hidden Condi-*

tional Random Fields, HCRFs), ou simplesmente CRFs com variáveis latentes, fornecem uma contrapartida discriminativa aos *classificadores* baseados em HMMs.

Como apresentado na seção 5.3.2, classificadores baseados em modelos de Markov associam a uma determinada sequência de observações $\boldsymbol{x}$ uma classe $\omega_i \in \Omega$ através da probabilidade condicional $p(\omega_i|\boldsymbol{x})$ estimada seguindo um critério MAP ou ML a partir da aplicação da regra de Bayes utilizando c modelos gerativos λ_i, $1 \leq i \leq c$. Como é possível notar, temos novamente um caso em que a estimação de um modelo mais geral ocorre como um passo intermediário na tentativa de resolução de um problema específico – a classificação.

Para o problema de associar um rótulo de classe ω_i a uma determinada sequência de observações $\boldsymbol{x}$ podemos, ao invés de aprender c modelos gerativos distintos para obter $p(\omega|\boldsymbol{x})$ através da regra de Bayes, aprender um único modelo discriminativo para obter $p(\omega|\boldsymbol{x})$ de maneira direta, como mostra a Figura 61. Para tanto, podemos considerar a probabilidade condicional modelada apresentada em (5.30) como ponto de partida, e então acrescentar os rótulos de classe $\boldsymbol{\omega}$ que queremos predizer.

A equação (5.30), por conveniência, é reproduzida a seguir:

$$\mathrm{p}(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathrm{Z}(\mathbf{x})} \prod_{\mathrm{C_p} \in \mathcal{C}} \prod_{\Psi_\mathrm{c} \in \mathrm{C_p}} \Psi_\mathrm{c}(\mathbf{x_c}, \mathbf{y_c}; \boldsymbol{\theta_p}) \,. \tag{5.43}$$

Expandindo esta equação para adicionar os rótulos de classe $\boldsymbol{\omega}$, obtemos

$$\mathrm{p}(\mathbf{y}, \omega|\mathbf{x}) = \frac{1}{\mathrm{Z}(\mathbf{x})} \prod_{\mathrm{C_p} \in \mathcal{C}} \prod_{\Psi_\mathrm{c} \in \mathrm{C_p}} \Psi_\mathrm{c}(\mathbf{x_c}, \mathbf{y_c}, \omega_\mathrm{c}; \boldsymbol{\theta_p}) \tag{5.44}$$

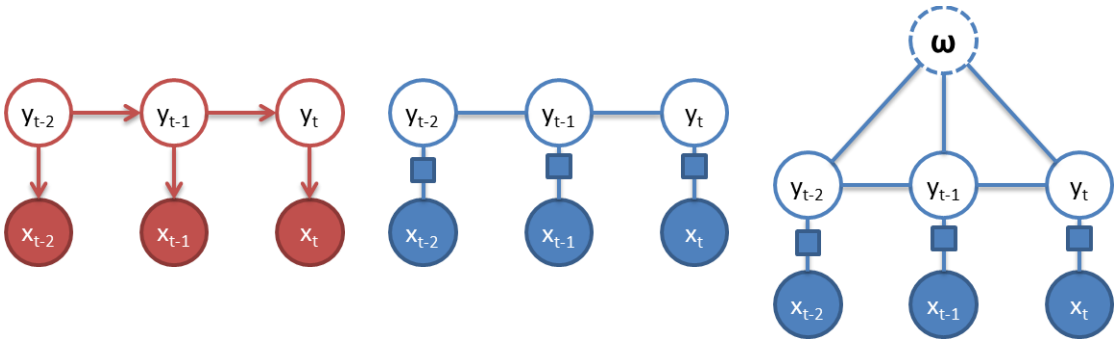

Figura 61. Comparação entre modelos gráficos de HMMs, CRFs e HCRFs, respectivamente. Pode-se notar que HMMs são modelos gerativos enquanto CRFs e HCRFs são modelos discriminativos. Em HCRFs, os rótulos y_t de cada observação são tratados como variáveis latentes (não observáveis).

em que $Z(\boldsymbol{x})$ é atualizada para considerar todos os possíveis rótulos $\boldsymbol{\omega}$, como

$$\mathrm{Z}(\mathbf{x}) = \sum_{\omega}\sum_{\mathbf{y}}\prod_{\mathrm{C_p}\in\mathcal{C}}\prod_{\Psi_\mathrm{c}\in\mathrm{C_p}}\Psi_\mathrm{c}(\mathbf{x_c},\mathbf{y_c},\omega_\mathrm{c};\ \boldsymbol{\theta_p}) \tag{5.45}$$

e cada fator Ψ_c também é estendido de maneira tal que possa ser escrito na forma

$$\Psi_\mathrm{c}(\mathbf{x_c},\mathbf{y_c},\omega_\mathrm{c};\ \theta_\mathrm{p}) = \exp\left\{\sum_{\mathrm{k}=1}^{\mathrm{K(p)}}\lambda_{\mathrm{pk}}\mathrm{f_{pk}}\ (\mathbf{x_c},\ \mathbf{y_c},\ \omega_\mathrm{c})\right\}. \tag{5.46}$$

A suposição de que os rótulos de sequência $\boldsymbol{y}$ não sejam mais observáveis durante o treinamento, tal como nos modelos ocultos de Markov, requer a marginalização sobre todas as possíveis sequências de estados ocultos $\boldsymbol{y}$ dando origem à função de verossimilhança marginal $p(\omega|\boldsymbol{x})$ denotada por

$$\mathrm{p}(\omega|\mathbf{x}) = \sum_{\mathbf{y}}\mathrm{p}(\mathbf{y},\omega|\mathbf{x})\,. \tag{5.47}$$

A primeira vista, esta soma pode parecer intratável de ser calculada, já que envolve uma somatória sobre todas as possíveis sequências de estados ocultos $\boldsymbol{y}$**.** A chave para se calcular esta soma de maneira eficiente é primeiro calcular uma distribuição sobre os possíveis estados ocultos $\boldsymbol{y}$ que o modelo poderá assumir para uma dada sequência $\boldsymbol{x}$ de rótulo ω

$$\mathrm{p}(\mathbf{y}|\mathbf{x},\omega) = \frac{\prod_{\mathrm{C_p}\in\mathcal{C}}\prod_{\Psi_\mathrm{c}\in\mathrm{C_p}}\Psi_\mathrm{c}(\mathbf{x_c},\mathbf{y_c},\omega_\mathrm{c};\ \boldsymbol{\theta_p})}{\sum_{\mathbf{y}'}\prod_{\mathrm{C_p}\in\mathcal{C}}\prod_{\Psi_\mathrm{c}\in\mathrm{C_p}}\Psi_\mathrm{c}(\mathbf{x_c},\ \mathbf{y}'_\mathbf{c},\omega_\mathrm{c};\ \boldsymbol{\theta_p})}. \tag{5.48}$$

Continuando, pode-se escolher $Z(\omega,\boldsymbol{x})$ tal que

$$\mathrm{Z}(\omega,\mathbf{x}) = \sum_{\mathbf{y}'}\prod_{\mathrm{C_p}\in\mathcal{C}}\prod_{\Psi_\mathrm{c}\in\mathrm{C_p}}\Psi_\mathrm{c}(\mathbf{x_c},\ \mathbf{y}'_\mathbf{c},\omega_\mathrm{c};\ \boldsymbol{\theta_p}) \tag{5.49}$$

o que permite obter a expressão

$$p(\mathbf{y}|\mathbf{x},\omega) = \frac{1}{Z(\omega,\mathbf{x})} \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \Psi_c(\mathbf{x_c}, \mathbf{y_c}, \omega_c; \boldsymbol{\theta_p}) . \tag{5.50}$$

Agora, a partir de (5.45) e (5.46), pode-se obter

$$\begin{aligned} p(\omega|\mathbf{x}) &= \sum_{\mathbf{y}} \left(\frac{1}{Z(\mathbf{x})} \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \Psi_c(\mathbf{x_c}, \mathbf{y_c}, \omega_c; \boldsymbol{\theta_p}) \right) \\ &= \frac{1}{Z(\mathbf{x})} \sum_{\mathbf{y}} \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \Psi_c(\mathbf{x_c}, \mathbf{y_c}, \omega_c; \boldsymbol{\theta_p}) \end{aligned} \tag{5.51}$$

e, finalmente, a partir de (5.48) e (5.50) pode-se mostrar que

$$p(\omega|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \sum_{\mathbf{y}} \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \Psi_c(\mathbf{x_c}, \mathbf{y_c}, \omega_c; \boldsymbol{\theta_p}) = \frac{Z(\omega,\mathbf{x})}{Z(\mathbf{x})} \tag{5.52}$$

Nesta expressão, pode-se notar que a constante de normalização $Z(\boldsymbol{\omega}, \boldsymbol{x})$ pode ser computada pelo mesmo algoritmo de inferência utilizado para computar a função partição $Z(\boldsymbol{x})$ dos CRFs originais (SUTTON e MCCALLUM, 2007) como os apresentados na seção 5.4.

5.5.1 Aprendizado

Assim como no caso de CRFs, o aprendizado em HCRFs se resume a resolver um problema de otimização irrestrita, que pode ser resolvido utilizando-se qualquer método de otimização disponível. No entanto, o preço que pagamos pela maior flexibilidade proporcionada pelos estados ocultos é a presença de mínimos locais neste problema de otimização. Isto quer dizer que o problema de otimização envolvido no aprendizado de HCRFs não é mais convexo e, portanto, apresenta o mesmo problema de múltiplos mínimos locais tal como no aprendizado de redes neurais apresentado na seção 5.1. Apesar desta desvantagem, podemos empregar o mesmo algoritmo de Retropropagação Resiliente, originalmente proposto para o aprendizado de redes neurais, para efetuar seu aprendizado; que como mostram Mahajan, Gunawardana e Acero (2006) se revelou um dos melhores algoritmos para este fim.

Supondo um problema de aprendizado supervisionado, novamente consideraremos um conjunto *i.i.d* de $\boldsymbol{N}$ instâncias de treinamento na forma

$$\mathcal{D} = \{\mathbf{x^{(i)}}, \omega^{(\mathbf{i})}\}, 1 \leq \mathrm{i} \leq \mathrm{N} \tag{5.53}$$

em que cada $\boldsymbol{x}^{(i)}$ seja um vetor de observações $\boldsymbol{x}^{(i)} = \langle x_1^{(i)}, x_2^{(i)}, \ldots, x_T^{(i)} \rangle$ e cada $\omega^{(i)}$ seja o rótulo de classe associado com o vetor de observações $\boldsymbol{x}^{(i)}$. Uma escolha adequada para a função objetivo a ser otimizada pode ser dada pela função de verossimilhança marginal dada por

$$\begin{aligned} \ell(\boldsymbol{\theta}) = \sum_{i=1}^{N} log\, p(\omega^{(i)}|\boldsymbol{x}^{(i)}) &= \sum_{i=1}^{N} log \frac{Z(\omega^{(i)}, \boldsymbol{x}^{(i)})}{Z(\boldsymbol{x}^{(i)})} \\ &= \sum_{\mathrm{i}=1}^{\mathrm{N}} \log \mathrm{Z}(\omega^{(\mathbf{i})}, \mathbf{x}^{(\mathbf{i})}) - \sum_{\mathrm{i}=1}^{\mathrm{N}} \log \mathrm{Z}(\mathbf{x}^{(\mathbf{i})}) . \end{aligned} \tag{5.54}$$

Novamente, podemos incorporar controle da capacidade através da adição de um termo de regularização na forma

$$\ell(\boldsymbol{\theta}) = \sum_{\mathrm{i}=1}^{\mathrm{N}} \log \mathrm{Z}(\omega^{(\mathbf{i})}, \mathbf{x}^{(\mathbf{i})}) - \sum_{\mathrm{i}=1}^{\mathrm{N}} \log \mathrm{Z}(\mathbf{x}^{(\mathbf{i})}) - \sum_{\mathrm{k}=1}^{\mathrm{K}} \frac{\lambda_{\mathrm{k}}^2}{2\sigma^2} . \tag{5.55}$$

Derivando-se $\ell(\boldsymbol{\theta})$ a respeito de todos os parâmetros $\boldsymbol{\lambda} \in \boldsymbol{\theta}$, obtemos o vetor gradiente $\boldsymbol{g} = \langle \frac{\partial \ell}{\partial \lambda_1}, \frac{\partial \ell}{\partial \lambda_2}, \ldots, \frac{\partial \ell}{\partial \lambda_K} \rangle$, $1 \leq k \leq K$, em que cada derivada parcial $\frac{\partial \ell}{\partial \lambda_k}$ é dada por

$$\begin{aligned} \frac{\partial \ell}{\partial \lambda_{\mathrm{pk}}} = &\sum_{\Psi_c \in C_p} \sum_{\boldsymbol{y}_c'} p(\boldsymbol{y}_c'|\omega, \boldsymbol{x})\, f_k(\omega_c, \boldsymbol{x}_c, \boldsymbol{y}_c') \; - \\ &\sum_{\Psi_c \in C_p} \sum_{\boldsymbol{y}_c'} \sum_{\boldsymbol{\omega}_c'} p(\boldsymbol{y}_c', \omega_c|\boldsymbol{x}_c) f_{pk}(\omega_c', \boldsymbol{x}_c, \boldsymbol{y}_c') - \frac{\lambda_{\mathrm{k}}}{\sigma^2} , \end{aligned} \tag{5.56}$$

que pode ser interpretado como a probabilidade de se seguir através de um caminho $\boldsymbol{y}_c'$ vezes o número de ocorrências da característica f_k durante o caminho. A diferença entre as duas probabilidades marginais pode igualmente ser interpretada como a probabilidade do modelo atual versus a probabilidade de um modelo total-

mente observado. É possível mostrar que estas são as mesmas probabilidades utilizadas na estimação pelo critério de Máxima Informação Mútua (*Maximum Mutual Information*, MMI) de HMMs no caso de HCRFs utilizando cadeias lineares (GUNAWARDANA, MAHAJAN, *et al.*, 2005).

Outro detalhe igualmente interessante ocorre ao notar que, utilizando a representação em cliques maximais, pode-se replicar exatamente a estrutura dos classificadores baseados em HMMs. A diferença reside apenas no fato dos parâmetros de um HCRF não estarem restritos a formar probabilidades, como ilustrado na Figura 62. Pode-se observar, assim, que o conjunto de soluções possíveis de serem obtidas por um classificador baseado em HMM se restringe a apenas um subconjunto do conjunto de soluções possíveis de serem expressas por um HCRF.

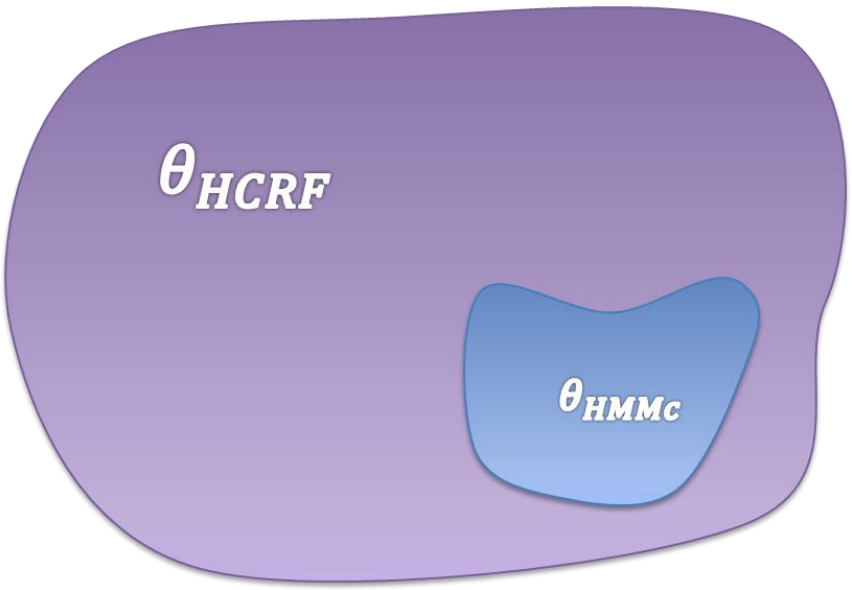

Figura 62. Representação do espaço de parâmetros para classificadores baseados em HMMs e HCRFs. Pode-se notar que todo classificador baseado em HMMs é também um HCRF, porém nem todo HCRF pode ser transformado em um classificador baseado em HMMs.

5.6 Resumo do capítulo

Neste capítulo, conhecemos várias das principais técnicas de aprendizado de máquina e reconhecimento de padrões potencialmente aplicáveis ao problema de reconhecimento de gestos. Após a detecção e segmentação dos membros superiores do articulador da Libras realizada pelos métodos de visão computacional do capítulo anterior, poderemos aplicar os classificadores aqui apresentados para reconhecer tanto observações de gestos estáticos, na forma de configurações de mãos em quadros isolados; quanto sequências destas observações.

No caso de quadros isolados, poderemos utilizar ANNs e SVMs; no caso de sequências destas observações, poderemos utilizar modelos classificadores de sequências como HMMs e HCRFs. No entanto, vimos neste capítulo diversas argumentações sustentando uma escolha criteriosa e bem fundamentada dos métodos mais adequados para este fim. Nos capítulos seguintes, destinados ao detalhamento da execução desta pesquisa em si, veremos de maneira concreta como cada um destes modelos se comporta quando confrontados com um problema real e cuja importância social destacamos muitas vezes ao decorrer deste trabalho: o reconhecimento dos sinais da Libras.

Parte II

Desenvolvimento

Capítulo 6

Abordagem proposta

"Science may be described as the art of systematic over-simplification – the art of discerning what we may with advantage omit." — Sir Karl Popper, 1992

Neste capítulo, apresentaremos a abordagem utilizada para combinar as ideias apresentadas nos capítulos anteriores na construção de um sistema para reconhecimento da Língua Brasileira de Sinais.

Como apresentado no Capítulo 1, vimos que o intérprete desempenha papel fundamental na divulgação da Libras e no trabalho de inclusão do surdo na sociedade brasileira. A ideia de se construir uma ferramenta capaz de interpretar a Libras pode prover a independência necessária ao surdo para, expressando suas ideias em língua de sinais, ganhar o poder de atingir não apenas a comunidade surda, mas também a comunidade ouvinte. A criação de tal ferramenta pode significar a derrubada de várias barreiras para a inclusão social do surdo. Ainda que o trabalho aqui apresentado não se torne uma solução concreta ou definitiva neste sentido, o autor deste trabalho julga que o caminho trilhado até aqui servirá de inspiração e que as soluções para cada problema intermediário aqui encontradas sirvam de alimento para futuros desenvolvimentos e esforços direcionados para este fim, complementando a literatura apresentada no Capítulo 2.

Vimos no Capítulo 3 que o problema de reconhecimento de um simples sinal em Libras depende da combinação simultânea de diversas características e que a sua interpretação final é altamente sensível ao contexto em que um sinal se insere devido a ambiguidade da língua. No Capítulo 4, vimos métodos e técnicas de processamento de imagens passíveis de serem aplicados na extração de características do sinal da Libras, e no Capítulo 5 vimos como as Máquinas de Vetores de Suporte (SVM) e os Campos Condicionais Aleatórios (CRF/HCRF) fornecem meios e técnicas para classificar e interpretar estes sinais.

6.1 Dividir-para-conquistar

A abordagem para o reconhecimento de palavras da Libras adotada nesta pesquisa é claramente inspirada nos avanços oriundos do campo de reconhecimento de fala, cuja comunidade pesquisadora têm se beneficiado de uma literatura sempre crescente durante as últimas décadas. Da mesma maneira que métodos de reconhecimento de fala frequentemente constroem modelos de linguagem sobre classificadores de fonemas, também aqui construiremos modelos baseados nas unidades básicas da Libras. Teremos aqui, portanto, duas camadas de reconhecimento; uma para extração de informação acerca destas unidades, e outra para contextualização e final classificação do sinal sendo articulado.

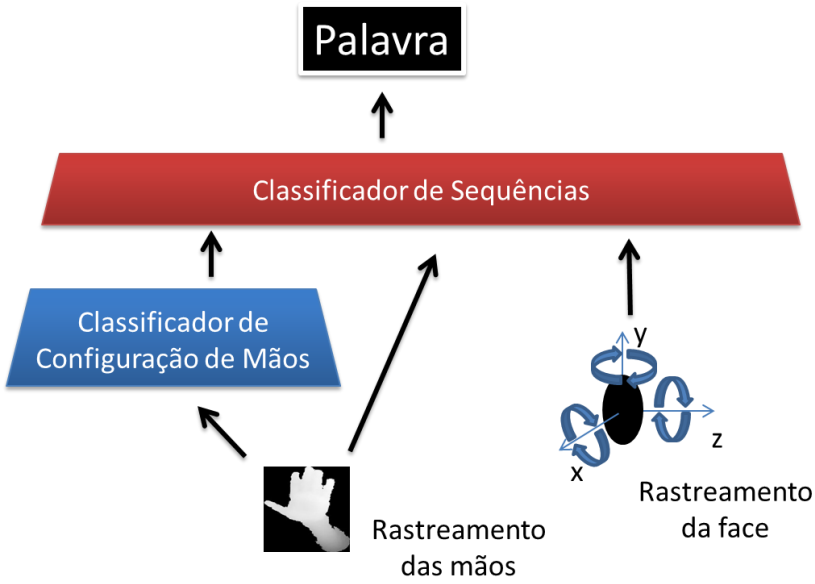

Figura 63. Arquitetura em duas camadas utilizadas no experimento envolvendo imagens de profundidade e palavras naturais da Libras.

Nesta abordagem, as duas camadas diferem no uso ou não da informação temporal durante a execução de um sinal. O papel da primeira camada é atuar como uma extratora de características para cada quadro capturado pelos sensores de luminosidade e profundidade, gerando uma representação simplificada da informação contida em um único quadro. No contexto de reconhecimento de gestos da Libras, a primeira camada será responsável por extrair informações acerca da configuração de mãos, da face e da posição relativa das mãos em relação à face do usuário.

Se a primeira camada de processamento é responsável pela extração de informações em quadros individuais, a segunda camada de processamento é responsável pela extração de informações de sequências destes quadros. Ou seja, esta camada deverá avaliar o contexto temporal em que os quadros estão inseridos, e a partir das informações coletadas pela primeira camada, gerar um rótulo de classe para cada palavra sendo articulada. Para a criação desta camada utilizaremos HMMs e HCRFs, realizando testes analisando as vantagens e desvantagens na utilização de cada um destes modelos para compor a segunda camada deste sistema.

6.2 Segmentação e extração de características

Para a extração de atributos a partir das sequências de imagens durante a classificação de gestos em imagens de profundidade, utilizamos a combinação dos métodos de *Viola-Jones* e *Camshift*, apresentados no Capítulo 4. Uma vez que a posição da face tenha sido estimada a partir da imagem colorida, podemos aproveitar a disponibilidade do mapa de profundidade do sensor *Kinect* para criar um algoritmo de segmentação adaptativo baseado no mapeamento da estimativa da face na imagem de profundidade.

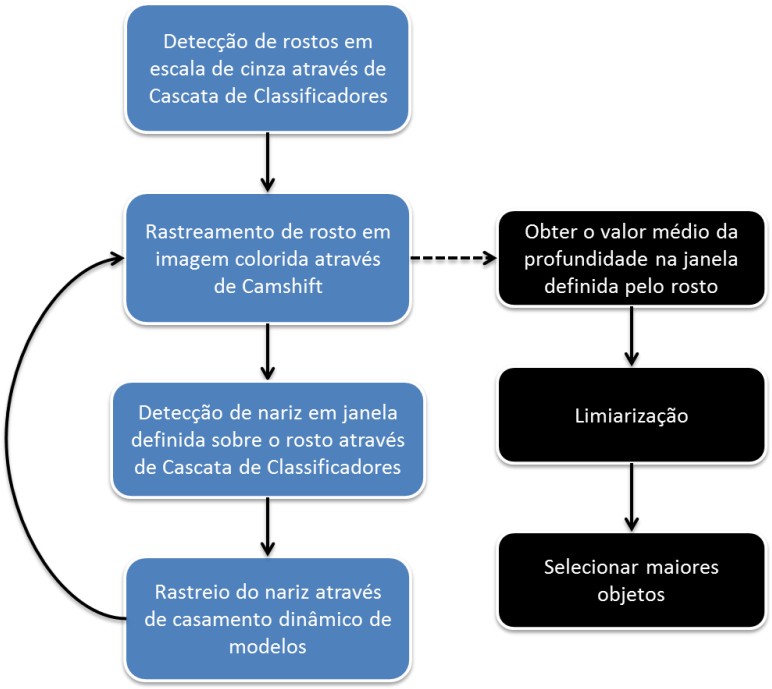

Figura 64. Esquematização do algoritmo de segmentação proposto neste trabalho, motivado e inspirado pelo trabalho de (ANJO, 2012).

Tal algoritmo de segmentação, representado na Figura 64, é capaz de extrair estimativas simples das formas e posições das mãos, bem como estimativas da posição e orientação da face (Figura 65). É possível notar que, uma vez que este algoritmo se baseia na localização da face, a oclusão torna-se um problema significativo em seu funcionamento. Este problema, porém, pode ser tratado ou amenizado utilizando-se um filtro dinâmico linear como um filtro de *Kalman* ou outras estratégias simples para detecção de mudanças bruscas na estimativa de profundidade da face, uma vez que, durante a articulação do sinal na Libras é pouco provável que o articulador esteja em movimento no eixo z representado na Figura 65. Este algoritmo se fundamenta nas ideias centrais defendidas por (ANJO, 2012).

Para a extração de características a partir das formas de mãos coletadas, reduziremos as formas de mãos para um espaço de 32×32, formando um vetor de características de 1024 posições, como exibido na Figura 66. O sucesso desta abordagem para extração de características será dependente principalmente na habilidade deste sistema em rapidamente detectar um erro de rastreamento e recuperar-se deste erro. Esta técnica se torna passível de funcionar mesmo ao se deparar com ambientes ruidosos cujos fundos não sejam controlados. Como veremos em breve, a elasticidade e a capacidade da segunda camada de processamento tolerar quadros erroneamente classificados será fundamental para que esta abordagem funcione adequadamente.

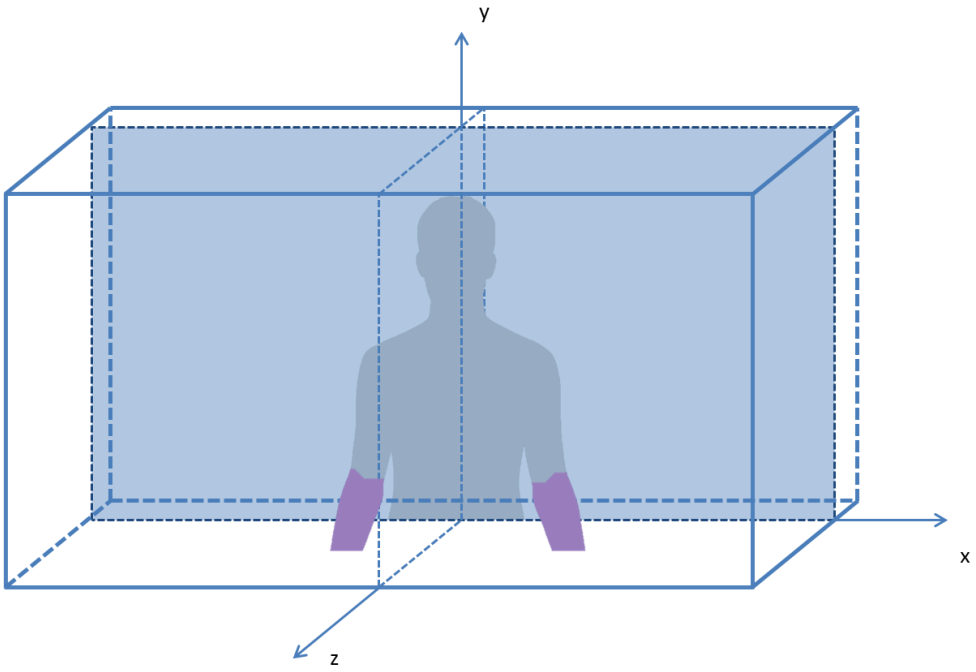

Figura 65. O espaço de sinalização segmentado utilizando o algoritmo apresentado.

Nos testes realizados, estaremos expressamente abdicando de características mais elaboradas de maneira a manter o reconhecimento por parte dos classificadores o mais direto possível; desta maneira, objetivaremos medir não apenas a capacidade de cada classificador em confrontar problemas com o mínimo de informação *a priori*, como também sua capacidade em lidar com problemas de alta dimensionalidade sem a necessidade de etapas especiais de pré-processamento ou redução de dimensionalidade.

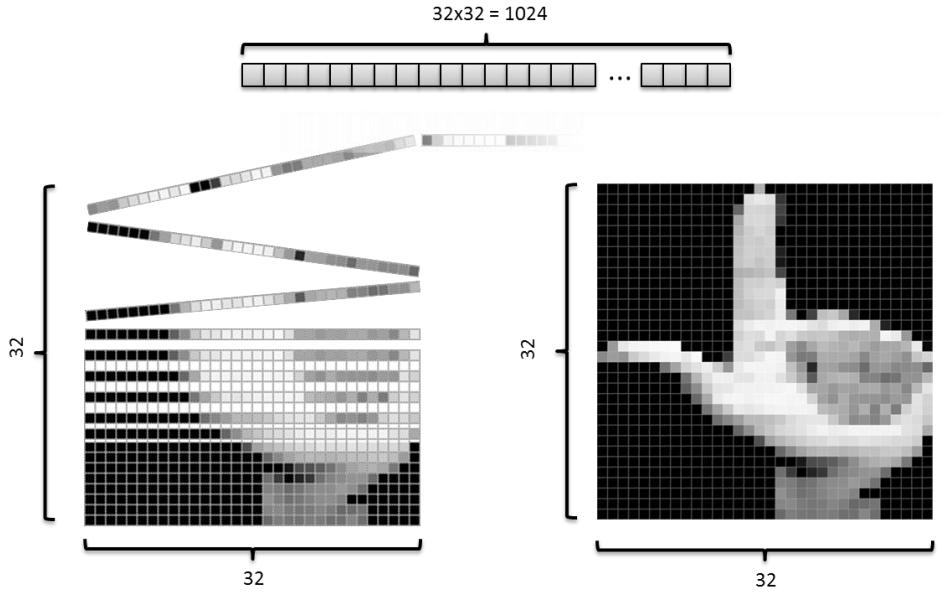

Figura 66. Extração do vetor de características a partir de um único quadro para alimentação da primeira camada de processamento.

Capítulo 7

Metodologia de pesquisa

"Art and science have their meeting point in method." — Edward Robert Bulwer-Lytto

Neste capítulo, apresentaremos a metodologia adotada para conduzir experimentos, testar hipóteses e extrair conclusões dos resultados obtidos durante esta pesquisa. Evidenciaremos como resultados obtidos a partir de diferentes métodos podem ser comparados de maneira objetiva através do uso de técnicas estatísticas e apresentaremos o delineamento geral do trabalho conduzido.

A fim de se evitar surpresas e tornar a execução desta pesquisa mais gerenciável, buscou-se primeiro validar a abordagem aqui proposta em um problema de menor complexidade, a ser apresentado na próxima seção. Detalharemos também como foi feita a aquisição dos dados de amostra utilizados nos experimentos realizados. Para a análise e comparação de resultados, adotamos medidas estatísticas quantitativas, que possibilitam a execução de testes de hipótese de forma a suportar os achados deste trabalho.

7.1 Validação da abordagem proposta

Sendo o problema de reconhecimento de palavras em Libras bastante complexo, antes de saltar diretamente na direção deste problema foi realizada a validação inicial da abordagem aqui proposta em um ambiente simplificado, cujos resultados foram publicados em (SOUZA, ANJO e PIZZOLATO, 2012). Este experimento focou no reconhecimento de palavras soletradas através do alfabeto manual em imagens capturadas em escala de cinza em um ambiente severamente controlado. No entanto, este experimento se mostrou de grande valia para certificar que todos componentes do sistema aqui proposto estivessem funcionando adequadamente; e também para fornecer a experiência necessária para a seleção de modelos.

7.2 Obtenção das amostras

Para possibilitar a criação e o aprendizado dos modelos de classificação nesta pesquisa, foi adquirido e organizado um banco de dados de sinais da Libras contendo poses, representadas na forma de imagens estáticas; e palavras naturais do vocabulário surdo, representadas na forma de sequências de imagens. Foram coletadas amostras de 21 participantes distintos, de ambos os sexos, que se voluntariaram para participar nesta pesquisa. Todos voluntários afirmaram ter lido e estarem de acordo com o termo de consentimento livre e esclarecido, documento este que lista todas as implicações decorrentes de sua participação. Os termos específicos assinados pelos participantes podem ser visualizados no Apêndice A.

A coleta de dados foi realizada através de um *software* criado especificamente para esta pesquisa, utilizando-se um sensor *Kinect* para registrar ambas as imagens coloridas e de profundidade. Durante sua participação, os voluntários permaneceram sentados, distantes cerca de 1 metro em frente ao sensor. O *software* utilizado empregava o algoritmo discutido no Capítulo 6, processando ao máximo de quadros por segundo suportado pelo sensor *Kinect* para imagens 640x480.

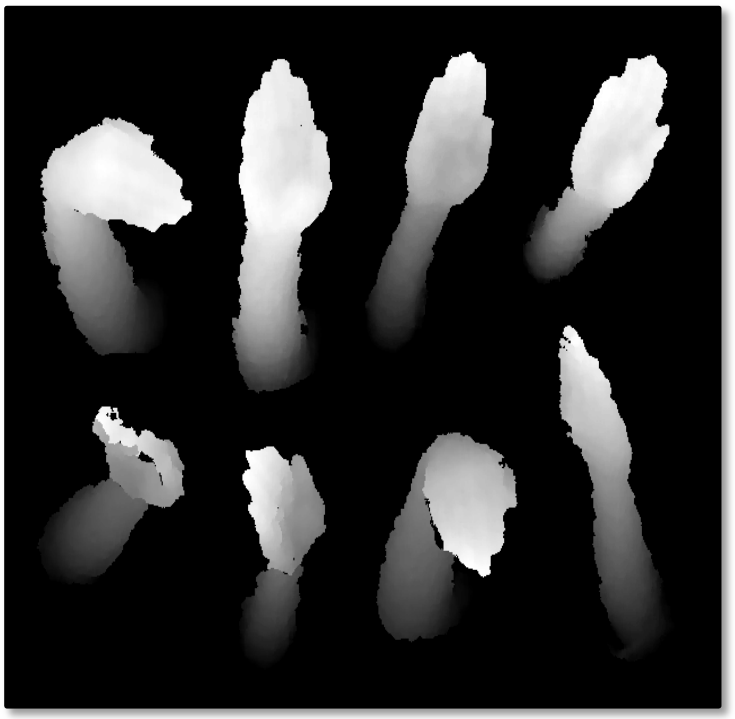

Figura 67. Diferentes variações de uma mesma configuração de mãos coletadas durante o experimento.

O processo de aquisição de dados ocorreu em duas etapas. Na primeira etapa, os participantes foram solicitados a realizar cada uma das 46 poses fundamentais da Libras, propositalmente variando a localização espacial e a orientação de sua mão dominante durante a gravação, ao passo em que mantinham sua configuração sempre fixa, conforme pode ser observado na Figura 67. Desta etapa, foi amostrado um total de 300 quadros para cada uma das possíveis 46 configurações de mãos, destinados para o treinamento dos classificadores, fornecendo um total de 13.800 amostras de treinamento. Outro conjunto independente e mutualmente exclusivo foi separado para servir de conjunto de validação nesta fase intermediaria de construção de nosso sistema.

Na segunda etapa da aquisição de amostras, solicitamos aos participantes que realizassem a articulação de 13 palavras do vocabulário natural da Libras. Estas palavras foram repetidas múltiplas vezes por cada participante, de maneira a acomodar pequenas variações entre as diferentes sinalizações. Isto nos forneceu 939 sequências de quadros e um total de 139.154 quadros para compor o banco de dados de gestos dinâmicos. Estas sequências foram divididas em 10 conjuntos mutualmente exclusivos em antecipação ao uso da validação cruzada em 10 partes.

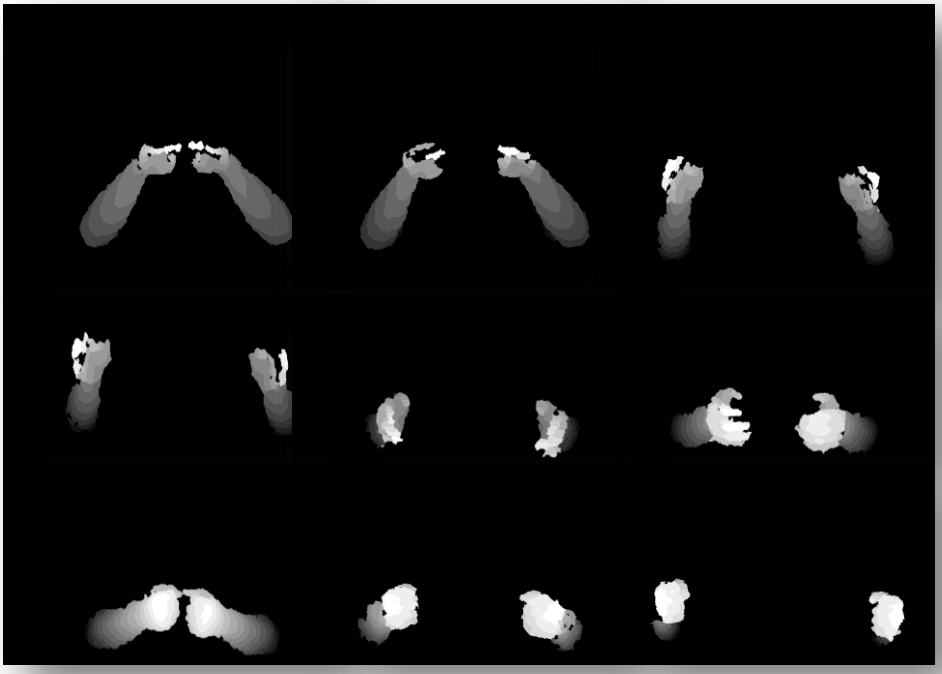

Figura 68. Execução da palavra Armário, coletada durante o experimento.

As palavras exploradas neste trabalho são exibidas na Tabela 2. Estas palavras foram escolhidas de acordo com as dificuldades que cada uma delas impõe ao sistema de reconhecimento: *Armário* envolve a oclusão da face, *Desculpa* envolve um movimento de inclinação com a cabeça. *Eu*, *Sapato*, *Gostar* e *Comprar* envolvem o toque em diferentes partes do corpo. *Desculpa* e *Idade* são articuladas com a mesma configuração de mãos, mas diferem em localização espacial e na presença do movimento de inclinação com a cabeça; *Armário* e *Carro* são articuladas dando igual importância a ambas as mãos, sem que haja uma mão dominante.

Tabela 2. Palavras selecionadas para compor o banco de gestos dinâmicos

Armário	Idade
Sapato	Carro
Desculpa	Comprar
Eu	Gostar
Dia	Nome
Tchau	Querer
Oi	

Esta escolha de palavras permitiu a verificação do algoritmo e da abordagem proposta sobre variadas dificuldades; porém deve-se notar que a falta de um banco de dados padronizado e apropriado para a condução de testes comparativos entre diferentes trabalhos acadêmicos ainda é uma lacuna a ser preenchida na literatura.

7.3 Análise de desempenho

Para avaliar o desempenho dos classificadores apresentados neste estudo, serão utilizadas medidas que permitam a comparação estatística dos dados. Um estudo comparativo de diversas medidas de desempenho aplicáveis para a análise qualitativa de resultados em classificação supervisionada é apresentado em (FERRI, HERNÁNDEZ-ORALLO e MODROIU, 2009). Neste trabalho, utilizaremos testes estatísticos para comparação de tabelas de contingência.

Considerando um problema de classificação entre k classes distintas, uma tabela de contingência (PEARSON, 1904), ou matriz de confusão, pode ser descrita por uma matriz $k \times k$ na forma

$$C = \begin{pmatrix} c_{i,j} & \cdots & c_{i,k} \\ \vdots & \ddots & \vdots \\ c_{k,j} & \cdots & c_{k,k} \end{pmatrix} \tag{7.1}$$

em que cada elemento c_{ij} denota o número de elementos da classe i classificados como sendo da classe j. Notoriamente, os elementos da diagonal denotam o número de classificações corretas, sendo que uma matriz estritamente diagonal denota uma classificação perfeita.

Uma vez obtida a matriz de contingência para o classificador em relação à verdade, pode-se derivar inúmeras medidas quantitativas para análise de desempenho, como o coeficiente *Kappa* (κ) de Cohen (1960) e o coeficiente *Tau* (τ) de Kendall (1938). Outras medidas incluem o coeficiente de contingência C de Pearson, a medida de associação T de Tschuprow (1939), a medida de associação V de Cramer (1946), o coeficiente Pi (π) de Scott (1955), a versão ajustada por Sakoda do coeficiente C, o coeficiente *Alfa* (α) de Krippendorff (1980) e os índices de concordância geral e de concordância atribuível ao acaso.

7.3.1 Coeficiente Kappa de Cohen

O coeficiente *Kappa*, primeiro proposto em (COHEN, 1960) para o caso de concordância entre dois avaliadores, é um método bastante razoável para avaliação de classificadores, tendo sido usado, por exemplo, para verificar a acurácia de classificadores pontuais em sensoriamento remoto em (CONGALTON, 1991) e imagens multiespectrais (LEVADA, 2010). Uma medida relacionada, porém distinta, é o coeficiente *Kappa* de Fleiss (1971), que é na realidade uma generalização do coeficiente π de Scott para o caso de múltiplos avaliadores.

Considerando-se a avaliação de n amostras, o coeficiente *Kappa* é dado por

$$\hat{\kappa} = \frac{p_0 - p_c}{1 - p_c} \tag{7.2}$$

em que p_{ij} é a razão de proporção das observações colocadas na célula c_{ij} da matriz de confusão, é dada por $p_{ij} = c_{ij}/n$, p_0 e p_c denotam a proporção de concordância observada e a proporção de concordância esperada ao acaso, respectivamente. Seus valores podem ser calculados como

$$p_0 = \sum_{i=1}^{k} p_{ii}\,, \qquad p_c = \sum_{i=1}^{k} p_{i+}p_{+i} \tag{7.3}$$

em que $p_{i\cdot}$ e $p_{\cdot j}$ denotam a probabilidade marginal dos elementos ao longo da linha i e a soma dos elementos ao longo da coluna j, respectivamente

$$p_{i+} = \sum_{i=1}^{k} p_{ij}\,, \qquad p_{+j} = \sum_{j=1}^{k} p_{ij}\ . \tag{7.4}$$

O valor de *Kappa* é restrito ao intervalo $[-1; +1]$, em que um valor igual a 1 denota uma perfeita concordância entre os avaliadores e um valor igual a -1 denota uma perfeita discordância. Um valor igual a 0 significa que a concordância entre os dois avaliadores pode ser explicada somente pelo acaso, e valores negativos significam que a concordância entre os dois avaliadores foi menor do que poderia ser simplesmente atribuída ao acaso. Para a computação do valor de *Kappa* diretamente a partir da matriz de confusão pode-se utilizar a fórmula apresentada em (CONGALTON, 1991), dada por

$$\hat{\kappa} = \frac{n\sum_{i=1}^{k} c_{ii} - \sum_{i=1}^{k} c_{i+}c_{+i}}{n^2 - \sum_{i=1}^{k} c_{i+}c_{+i}}. \tag{7.5}$$

Para determinar se o valor de Kappa é significantemente diferente de zero, ou seja, que o nível de concordância entre os dois avaliadores é significantemente diferente do que seria atribuível somente ao acaso, pode-se assumir que $\hat{\kappa}$ seja aproximadamente normal e computar a estatística z_0 dada por

$$z_0 = \frac{\hat{\kappa}}{\sqrt{\widehat{\mathrm{Var}}_0(\hat{\kappa})}}\,. \tag{7.6}$$

No caso de muitas amostras, é possível calcular a variância do índice *Kappa* através da formula apresentada em (FLEISS, COHEN e EVERITT, 1969):

$$\widehat{\mathrm{Var}}_0(\hat{\kappa}) = \frac{p_c + p_c^2 - \sum_{i=1}^{k} p_{i+}p_{+i}(p_{i+} + p_{+i})}{n(1-p_c)^2}. \tag{7.7}$$

Comparando-se esta estatística contra uma distribuição normal padrão, é possível rejeitar a hipótese de que o nível de concordância seja melhor do que apenas atribuível à chance se z_0 for suficientemente grande considerando-se um teste de única cauda (FLEISS, LEVIN e PAIK, 2003, p. 605). No entanto, esta estimativa não é a mais adequada em todos os contextos, como por exemplo, nos estudos da confiança de sistemas de avaliação (BANERJEE, CAPOZZOLI, *et al.*, 1999), já que pode ser utilizada apenas para considerar a hipótese de que o valor estimado $\hat{\kappa}$ seja diferente do que seria esperado de uma classificação devida somente ao acaso. Para comparar o valor estimado $\hat{\kappa}$ contra algum valor κ diferente de zero, deve-se utilizar a estimativa da variância proposta por Fleiss, Cohen e Everitt (1969)

$$\widehat{\mathrm{Var}}(\hat{\kappa}) = \frac{A + B - C}{n(1-p_c)^2} \tag{7.8}$$

em que

$$\begin{gathered} A = \frac{1}{n}\sum_{i=1}^{k} p_{ii}\left(1 - (p_{i\cdot}+p_{\cdot i})(1-\hat{\kappa})\right)^2, \\ B = (1-\hat{\kappa})^2 \sum_{i\neq j}^{k} p_{ij}(p_{i\cdot}+p_{\cdot i})^2, \qquad C = \left(\hat{\kappa} - p_c(1-\hat{\kappa})\right)^2 \end{gathered} \tag{7.9}$$

O teste então pode ser considerado utilizando-se a estatística

$$z = \frac{|\hat{\kappa} - \kappa|}{\sqrt{\widehat{\mathrm{Var}}_0(\hat{\kappa})}}\ . \tag{7.10}$$

Novamente, comparando-se esta estatística contra uma distribuição normal padrão pode-se rejeitar a hipótese de que o valor estimado $\hat{\kappa}$ seja diferente do valor κ caso a estatística z seja significantemente grande. Uma observação importante que deve ser feita a respeito do uso de *Kappa* é que seu valor, quando reportado

sozinho, não é suficiente para caracterizar uma boa medida de desempenho, pois não há uma maneira concreta de se interpretá-lo fora do contexto de um teste estatístico. Uma tabela para a interpretação dos possíveis valores de *Kappa* foi proposta em (LANDIS e KOCH, 1977). Apesar dos autores advertirem a arbitrariedade dos valores sugeridos, e a validade desta interpretação ser discutível, estes valores acabaram se tornando padrão ao decorrer da literatura (BANERJEE, CAPOZZOLI, *et al.*, 1999; EUGENIO, 2000).

Capítulo 8

Experimentos e resultados

"In science the credit goes to the man who convinces the world, not to the man to whom the idea first occurs." — Sir Francis Darwin

NESTE CAPÍTULO, EXPLORAREMOS os experimentos realizados neste trabalho, de maneira a prover uma verificação prática dos conceitos apresentados até aqui, bem como sustentar a subsequente apresentação e validação das conclusões resultantes desta pesquisa.

No Capítulo 7, apresentamos como os experimentos desta pesquisa foram divididos em duas fases. A primeira fase visou avaliar a viabilidade da abordagem proposta e da metodologia em uma versão simplificada do problema de reconhecimento de gestos da Libras. A segunda fase constituiu a execução dos experimentos propriamente necessários à concretização desta pesquisa. A seguir, nas próximas duas seções, apresentaremos ambos os experimentos, respectivamente.

8.1 Soletração simplificada em escala de cinza

Nesta seção, apresentamos o primeiro experimento conduzido para validar a abordagem apresentada neste documento. A partir do banco de dados de palavras soletradas em Libras coletado por Pedroso para uso em sua investigação (PIZZOLATO, ANJO e PEDROSO, 2010), realizamos uma versão preliminar da pesquisa proposta. Diferente do trabalho de Pizzolato e colegas, esta investigação avaliou o uso de SVMs e HCRFs em comparação aos métodos baseados em ANN e HMMs. Este experimento também serviu para sanar quaisquer impossibilidades técnicas ou empecilhos que poderiam surgir no decorrer do desenvolvimento da pesquisa. Além do mais, ao notar-se que os sinais da datilologia são também membros integrantes do conjunto de todas as possíveis configurações de mãos da Libras,

esta investigação pode-se revelar útil acerca da aplicabilidade dos modelos SVM na classificação dos gestos estáticos.

Este experimento se baseou na mesma decomposição em camadas discutida no Capítulo 6. Utilizamos uma primeira camada contendo classificadores capazes de operar em quadros individuais de uma sequencia de imagens, e uma segunda camada capaz de tomar as classificações oriundas da primeira camada como símbolos de um alfabeto discreto e então produzir novas classificações indicando a palavra sendo articulada, conforme apresentado na Figura 69.

Figura 69. Exemplo de classificação da palavra soletrada 'Pato' no primeiro experimento realizado.

8.1.1 Camada de reconhecimento de gestos estáticos

Os gestos contidos no banco de dados de Pedroso foram pré-processados utilizando-se um filtro da mediana, de maneira a se obter redução de ruído; um filtro de limiariazação de Otsu (1979), para a posterior extração do maior objeto da imagem; e finalmente o redimensionamento da amostra, de forma que todas as amostras das mãos tivessem um tamanho uniforme contido em uma janela de 32×32 pixels. Ao invés do realizado no trabalho de Pizzolato e colegas, as amostras foram mantidas em escala de cinza, e não em forma de silhueta, de maneira a manter a maior quantidade de informação possível nas imagens. A seguir, foram criados e avaliados classificadores baseados em ANN e SVM para rotulagem de quadros e classificadores baseados em HMMs e HCRFs para rotulagem das sequências.

Foram separadas 300 amostras de cada uma das 27 classes, escolhidas de maneira aleatória, resultando em um total de 8100 amostras. Note que, anterior ao processamento, uma fase de seleção de amostras foi realizada de maneira a remover transições espúrias entre os gestos, bem como erros de segmentação e de borramento. Realizamos testes considerando SVMs com funções kernel polinomial e Gaussiana. Para o parâmetro C, utilizou-se inicialmente o valor sugerido pela heurística citada na seção 5.2.5.

O tempo de avaliação pode se tornar uma desvantagem significante quando comparamos SVMs contra outros métodos de classificação. Em sua formulação padrão, não existem limitantes superiores no número de vetores de suporte (*Support Vectors*, SVs) requeridos para resolução de um problema. Como o número de SVs muitas vezes cresce linearmente com o número de amostras no conjunto de treinamento, isto rapidamente se torna problemático. Assim, conduzimos experimentos visando medir o impacto deste problema e checar a validade do modelo de decisão por DDAGs em superar esta limitação.

Este experimento considerou três escolhas de funções kernel: funções Gaussianas, Quadráticas, e Lineares. Para o kernel Gaussiano, uma busca em grade grossa-para-fina foi realizada no espaço de hiperparâmetros para se terminar um melhor parâmetro σ^2. Para cada máquina treinada, avaliamos o conjunto de validação duas vezes: uma vez utilizando a estratégia de votação *um-contra-um*, e posteriormente utilizando a decisão por DDAGs. Anotamos o desempenho de todos classificadores, medidos em termos do coeficiente Kappa (κ) de Cohen, o número total de vetores de suporte necessários para a estratégia de votação e o numero médio de avaliações de vetores de suporte encontrados pelo caminho de votação do DDAG. Como as máquinas lineares podem ser escritas em forma compacta, para máquinas lineares consideramos o número de avaliações de máquinas ao invés do número de vetores de suporte. Todas SVMs foram ensinadas utilizando-se o algoritmo de Otimização Mínima Sequencial (*Sequential Minimal Optimization*, SMO) de Platt (1998). Pontos iniciais para o kernel Gaussiano foram identificados com o auxilio do valor heurístico para σ sugerido em (CAPUTO, SIM, *et al.*, 2002), baseado no intervalo interquartil das estatísticas de norma d o conjunto de treinamento.

Para comparação, avaliamos também o uso de ANNs treinadas através do algoritmo de Retropropagação Resiliente. No entanto, se o tempo de avaliação é uma preocupação para as SVMs, o comportamento de treinamento é talvez uma das maiores preocupações para a criação de ANNs. Como seus algoritmos de aprendizado muitas vezes têm de lidar com múltiplos mínimos locais, o uso de heu-

rísticas e outros mecanismos de controle se tornam praticamente mandatórios para garantir um bom aprendizado. O objetivo deste experimento também foi mensurar o impacto de heurísticas de inicialização, especificamente o método de Nguyen-Widrow (1990), em problemas de alta dimensionalidade. Redes neurais *feed-forward* de única camada intermediária foram criadas variando-se o número de neurônios na camada oculta, numa tentativa de aplicar o controle da capacidade. Todas as redes foram treinadas até a convergência de seu erro mínimo quadrado.

Iniciando com a máquina de vetor de suporte com kernel Gaussiano, encontramos um comportamento similar ao descrito em (VALENTINI e DIETTERICH, 2004). Mais especificamente, encontramos que o valor de C não exerceu tanta influencia no desempenho do classificador quanto uma escolha apropriada para σ^2. Tanto a estatística κ e o número médio de vetores de suporte (SVs) para cada classificador resultante foi encontrado ser mais dependente em σ^2 do que em C.

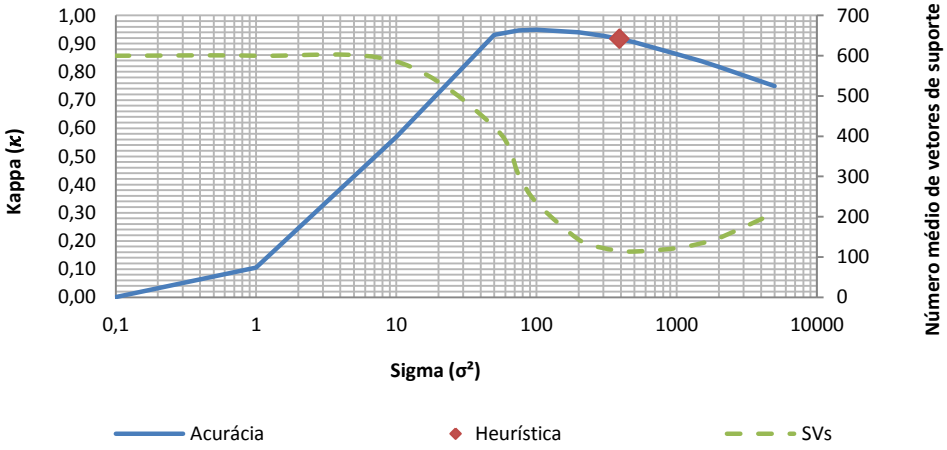

Figura 70. Cortes da busca em grade para C fixo no experimento em escala de cinza utilizando uma máquina de vetor de suporte com kernel Gaussiano.

A Figura 70 (acima) mostra um corte da busca em grade para $C = 1$ e variações de σ^2, juntamente com o número médio de vetores de suporte nas máquinas. Pode-se ressaltar que a escolha heurística para σ^2 como proposta por Caputo, Sim, *et al.* (2002) não apenas resultou em um bom desempenho no geral, mas também resultou no menor número de SVs, levando a soluções mais esparsas. Ainda que não tenha levado ao melhor κ possível, a heurística proveu um bom balanço entre acurácia e eficiência para os classificadores testados.

As superfícies de hiperparâmetros tanto em termos de κ quanto de esparsidade são exibidas na Figura 71. Destes gráficos pode-se notar uma relação quase direta entre o número de vetores de suporte e o desempenho de generalização; mais especificamente, há uma aparente correlação inversa entre o número de SVs e κ. De certa maneira, este comportamento pode parecer intuitivo, já que um maior número de SVs pode potencialmente levar a um aumento no overfitting e, portanto, a degradação de desempenho.

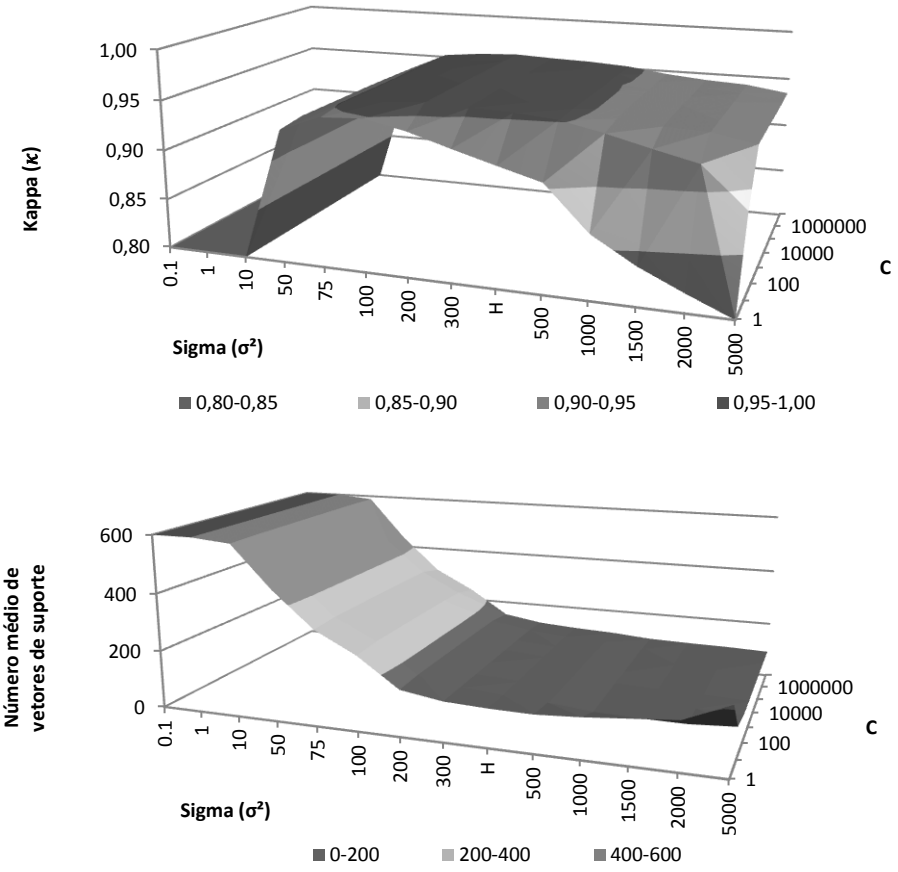

Figura 71. Plotagem das superfícies de desempenho (Kappa) e esparsidade (numero médio de vetores de suporte) para máquinas com kernel Gaussiano no experimento em escala de cinza.

Neste problema específico, o valor de C mostrou pouca ou nenhuma influencia no numero de SVs, exceto para altos valores de σ^2. Neste limite, um aumento em C parece contrabalancear um aumento em σ^2, levando novamente a um platô de esparsidade centrado na linha heurística ($H \cong 391{,}52$). Os gráficos não mostram valores para C maiores que 10^8 pois estes valores resultaram na divergência do algoritmo de aprendizado. Isto é compreensível, já que para altos valores de C a SVM de margem suave se aproxima da decisão de margem rígida, reduzindo sua habilidade de lidar com erros de classificação no conjunto de treinamento.

Após os testes com as funções kernel Gaussianas, o próximo passo foi medir as funções *kernel* Lineares e Quadráticas. Nota-se que o problema explorado neste experimento possui um alto número de dimensões de entrada, o que de certa forma torna esperado, portanto, que estes *kernels* produzissem bons resultados. Esta suspeita foi confirmada, como exibido na Figura 72. E, ainda que não exibido, não houve diferenças significativas entre o uso das versões homogêneas e não-homogêneas destas funções.

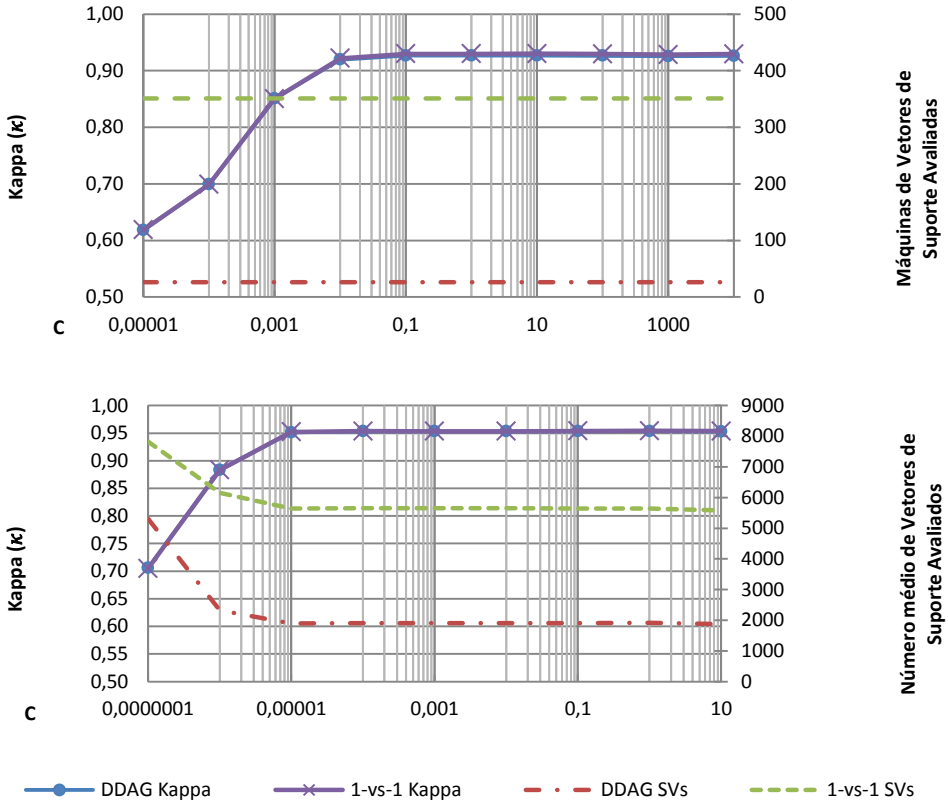

Figura 72. Desempenho das máquinas de vetores de suporte para funções kernel Lineares (acima) e Quadráticas (abaixo) no experimento em escala de cinza.

Os gráficos da Figura 72 também proveem uma comparação de eficiência entre as funções kernel Lineares e Quadráticas em termos do número de chamadas a função kernel e a avaliação de cada máquina. Como máquinas lineares podem ser escritas em forma compacta, seu tempo de avaliação é independente de seu numero de vetores de suporte, mas sim no número de máquinas na abordagem para múltiplas classes. As comparações de desempenho e esparsidade para as melhores máquinas encontradas são exibidas na Tabela 3.

Tabela 3. Melhores máquinas de vetores de suporte encontradas no experimento com escala de cinza.

Função kernel	Estratégia de Decisão	Validação	Número de Vetores de Suporte	
		Kappa ± (0.95 C.I.)	*Total*	*Médio*[1]
Linear	DDAG	0,9268 ± 0,006	27.602	1.527,34
	1-vs-1	0,9300 ± 0,006		5.483,00
Quadrática	DDAG	0,9536 ± 0,005	37.341	1.912,67
	1-vs-1	0,9542 ± 0,004		5.638,00
Gaussiana	DDAG	0,9586 ± 0,004	52.220	2.518,67
	1-vs-1	0,9583 ± 0,004		6.188,00

[1] Média de avaliações únicas por vetor de suporte ao classificar o conjunto de instâncias de validação.

A Tabela 3 mostra uma redução expressiva no numero médio de avaliações de SVs realmente necessárias durante a computação da decisão por DDAG e pela estratégia de votação, em comparação com o numero total de SVs em cada máquina multiclasse. Como a abordagem por votação necessita da avaliação de $c(c-1)/2 = 351$ máquinas a cada decisão, seu numero médio de SVs é sempre igual o numero total de vetores únicos encontrados durante o treinamento. O DDAG, por outro lado, precisa de um numero muito menor, ainda que muito variável, para alcançar uma decisão, dependendo somente na avaliação de $(c-1) = 26$ máquinas.

Concluindo-se os testes com SVMs, partimos para os resultados envolvendo ANNs. As redes encontradas se mostraram aptas a fornecer taxas de desempenho similares as SVMs. No entanto, esta habilidade veio anexada a um alto custo computacional para seu treinamento, especialmente ao considerar os custos em realizar múltiplas reinicializações aleatórias para se alcançar um bom mínimo local. Os melhores valores para κ foram encontrados entre 300 a 500 neurônios na camada intermediara, no mesmo intervalo encontrado em (PIZZOLATO, ANJO e PEDROSO, 2010). No entanto, podemos ver como o melhor desempenho alcançado por uma ANN ($\kappa = 0{,}9248$) foi bastante similar a uma SVM linear ($\kappa = 0{,}9268$).

Tabela 4. Desempenho das redes neurais feed-forward na classificação de quadros utilizando a Retropropagação Resiliente no experimento com escala de cinza.

Algoritmo	Inicialização	Validação	Neurônios na camada oculta
		Kappa ± (0.95 C.I.)	
RProp	Uniforme	0,9112 ± 0,006	500
RProp	Nguyen-Widrow	0,9248 ± 0,006	400

Dos resultados encontrados, pode-se notar que mesmo que a inicialização de Nguyen-Widrow (1990) não tenha afetado em demasia o desempenho final de redes pequenas, o mesmo não pode ser dito de redes com um maior número de neurônios. Redes com 700 neurônios ou mais não conseguiram aprender certas classes, apresentando uma taxa de classificação de zero para várias das classes do problema. Redes treinadas utilizando-se a inicialização de Nguyen-Widrow apresentaram resultados multo mais estáveis em comparação.

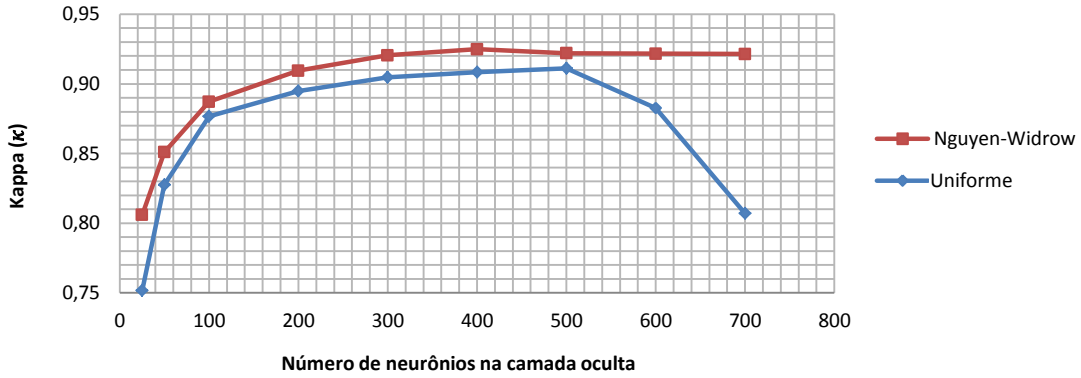

Figura 73. Gráfico comparativo do desempenho das redes neurais com e sem o uso de heurísticas de inicialização no experimento com escala de cinza.

O uso da inicialização de Nguyen-Widrow se mostrou extremamente útil para se alcançar um bom mínimo local em redes com um grande número de neurônios intermediários. Sem esta inicialização, as redes muitas vezes convergiam rapidamente para um mínimo local ruim após apenas um pequeno número de iterações (computacionalmente intensivas).

O uso da técnica de inicialização também mostrou prevenir à tão comum degradação associada com redes com um número de neurônios maior do que o necessário (Figura 73). E, ainda que não mostrado, pode-se ressaltar o fato da heurística ter auxiliado redes menores alcançar convergência bastante rapidamente, com um número bastante diminuto de iterações.

Ademais, podemos ressaltar que a melhor SVM encontrada ($\hat{\kappa}_{SVM} = 0{,}9586, \widehat{var}(\hat{\kappa}_{SVM}) = 5{,}098 \times 10^{-6}$) mostrou melhores resultados que a melhor ANN ($\hat{\kappa}_{ANN} = 0{,}9248, \widehat{var}(\hat{\kappa}_{ANN}) = 9{,}02 \times 10^{-6}$). Calculando a estatística

$$z = \frac{|\hat{\kappa}_{SVM} - \hat{\kappa}_{ANN}|}{\sqrt{\widehat{Var}_0(\hat{\kappa}_{SVM}) + \widehat{Var}_0(\hat{\kappa}_{ANN})}} \tag{8.1}$$

obtemos o valor de 8,995, ao passo que um teste de significância dos índices *Kappa* revela que a diferença entre os dois valores é significante a um nível de significância de 0,05. Assim, seria pouco provável que esta diferença de resultados pudesse ser atribuída somente ao acaso.

Caso a escolha do melhor classificador fosse baseada apenas em seu valor estimado de κ, esta escolha claramente resultaria na seleção de uma máquina SVM Gaussiana ou Polinomial. No entanto, estas escolhas não seriam muito praticas de um ponto de vista computacional devido ao alto numero de SVs que deveriam ser mantidos e computados sempre que uma nova decisão precisasse ser feita. A decisão por DDAGs, em contrapartida, proporciona uma melhora considerável no tempo de processamento, mas ainda assim requer um número alto e possivelmente altamente variável de SVs avaliados a cada decisão. Em aplicações em que um tempo constante de avaliação seja desejável, esta característica pode tornar a escolha de DDAGs pouco atrativa.

O uso de ANNs ao invés das SVMs listadas no parágrafo acima traz muitas vantagens. Entre elas, podemos enunciar que as ANNs desempenham bem esta tarefa com um número moderado de neurônios ocultos, as tornando aptas para processamento em tempo hábil (PIZZOLATO, ANJO e PEDROSO, 2010). No entanto, de uma perspectiva de aprendizado, existem muitas desvantagens nesta escolha. A correta inicialização dos pesos de uma ANN parece ser crítica para se alcançar um bom mínimo local. Ainda mais, os elevados tempos de treinamento podem dificultar ou mesmo impossibilitar tanto o ajuste correto dos hiperparâmetros de aprendizado bem como o uso de técnicas de estimação de desempenho mais elaboradas, como a validação cruzada em k-partes e o *bootstraping*.

Considerando todas as opções avaliadas, a escolha de melhor compromisso se apresenta na forma de um DDAG composto por SVMs lineares. Este modelo consegue prover tempos de avaliação comparáveis as ANNs e, ao mesmo tempo, oferecer a facilidade um algoritmo de aprendizado convexo. Além do mais, DDAGs baseados em SVMs lineares possuem um tempo de avaliação constante já que sua avaliação não depende do numero de vetores de suporte em cada máquina. Como um DDAG-SVM linear reduz o esforço de se computar 351 máquinas para apenas 26 decisões de tempo constante, pode-se obter um classificador altamente eficiente, ainda que moderadamente eficaz, a ser utilizado como a primeira camada do sistema de reconhecimento de gestos envisionado.

8.1.2 Camada de reconhecimento de gestos dinâmicos

Após a realização dos testes para a escolha de modelos SVM e ANNs adequados, os melhores modelos encontrados foram utilizados para realizar a rotulação de todo o banco de palavras contido no conjunto de dados. Este banco contem um vocabulário de 15 palavras diferentes articuladas utilizando-se o alfabeto manual da Libras, cobrindo 19 de um total de 27 letras possíveis. As palavras coletadas e organizadas por Pedroso são exibidas na Tabela 5.

Tabela 5. Palavras coletadas por Pedroso, 2010.

Arara	Cisne	Cobra	Foca	Gato
Jaguatirica	Leopardo	Macaco	Pato	Perereca
Peru	Rato	Sapo	Tatu	Urso

Exemplos do resultado da rotulação das amostras pertencentes a cada sequência de quadros podem ser vistos na Figura 74. Como se pode observar, a classificação obtida apresenta diversos erros. Este comportamento, apesar de a primeira vista parecer prejudicial à próxima etapa de classificação, poderá ser facilmente contornado utilizando-se a hipótese de Markov. As amostras incorretamente rotuladas poderão ser ignoradas, já que os HMMs, e, consequentemente, os HCRFs, são capazes de apresentar certa elasticidade temporal, bem como robustez no caso de rotulagens incorretas em meio à sequência de observações.

Figura 74. Rotulações dos quadros de diferentes sequências de imagens correspondentes à articulação das palavras Pato, Arara e Macaco em alfabeto manual.

Na Figura 74, também podemos observar que, na presença de erro de classificação de amostras individuais, esta classificação incorreta tende a estar próxima de letras equivalentes ou bastante similares, o que permitirá que estes erros sejam contornados através do contexto da sequência.

Uma vez que o banco tenha sido inteiramente rotulado, foram criados modelos baseados em HMM e HCRFs. O modelo de classificação HMM foi criado utilizando-se o critério MAP a partir do método de Bayes empírico. O aprendizado de cada modelo de Markov para cada classe de sequências foi realizado utilizando-se o algoritmo de *Baum-Welch*. Estes modelos serviram de base para inicializar os HCRFs a partir da escolha de funções características equivalentes a de um classificador baseado em HMM. Para o treinamento dos HCRFs, utilizamos o mesmo algoritmo Rprop utilizado para Redes Neurais apresentado na seção 5.1.2. O trabalho de Mahajan, Gunawardana e Acero (2006) mostra que o emprego do Rprop é competitivo com o uso do Gradiente Descendente Estocástico, sendo porém preferível dada a sua maior robustez a escolha de parâmetros de aprendizado. Em todas as instancias, o desempenho destes classificadores foi testado utilizando-se a técnica de validação cruzada em 10 partes. Os resultados são apresentados na Tabela 6 a seguir.

Tabela 6. Resultados para classificação de palavras utilizando-se HMMs e HCRFs mensurados através de validação cruzada em 10 partes, no experimento com escala de cinza.

		Treinamento		Validação	
Rotulação	**Classificação**	**Kappa**	**Variância**	**Kappa**	**Variância**
SVM	HMM	0,94686	1,112E-04	0,81922	2,960E-03
SVM	**HCRF**	**0,98368**	**3,539E-05**	**0,83317**	**2,787E-03**
ANN	HMM	0,94818	1,085E-04	0,80352	3,126E-03
ANN	HCRF	0,99030	2,115E-05	0,82358	2,830E-03

O valor de *Kappa* final foi calculado a partir da média dos valores *Kappa* para cada rodada de validação. De maneira similar, a variância apresentada junto aos valores de *Kappa* foi estimada através do cálculo da variância agrupada (*pooled variance*) para cada rodada. Pode-se observar que o melhor resultado obtido foi encontrado utilizando-se a rotulagem por SVMs e HCRFs (SVM+HCRF). No entanto, cabe investigar aqui se este resultado poderia ser atribuído somente ao acaso.

Para averiguar a hipótese de que o resultado apresentado pelo modelo SVM+HCRF é realmente diferente dos valores encontrados pelos demais modelos foram novamente realizados testes de significância para os valores *Kappa*. Os testes revelam que, para o conjunto de validação, a diferença entre os resultados dos modelos SVM+HCRF e SVM+HMM não é significante a um nível de significância para $p < 0.05$. Entretanto, há uma significativa diferença entre os valores para o conjunto de treinamento. Estes resultados motivam a interpretação de que os modelos baseados em HCRF são capazes de apresentar taxas de classificação significantemente maiores para um conjunto de dados e ainda apresentar níveis de generalização equivalentes ou marginalmente melhores do que os modelos baseados em HMMs. Em outras palavras, os modelos discriminativos mostraram-se capazes de absorver maior conhecimento sobre os dados apresentados sem que isso causasse *overfitting*.

8.1.3 Conclusão deste experimento

Neste experimento, detalhamos nossas descobertas iniciais no uso de SVMs e HCRFs em comparação a ANNs e HMMs na tarefa de reconhecimento de palavras soletradas em Libras, alcançando resultados estatisticamente significativos. Apesar da significância estatística e melhor acurácia dos modelos baseados em SVM, este experimento também revelou que a escolha apropriada do classificador de sequências para compor a segunda camada de processamento incumbia em um impacto muito maior do que qualquer escolha particular de classificador de quadros. Fara cada escolha possível, a utilização de HCRFs ao invés de HMMs resultou em uma maior capacidade para a retenção de informação de treinamento, ao mesmo tempo em que manteve a habilidade de generalização de nosso sistema intacta.

8.2 Palavras naturais em imagens de profundidade

Neta seção, apresentamos o experimento conduzido utilizando-se o banco de dados apresentado na seção 7.2, reunindo todos os aspectos desta pesquisa conforme apresentado no Capítulo 6. Como proposto, utilizamos a arquitetura em duas camadas, composta por classificadores de gestos estáticos e classificadores de gestos dinâmicos.

A primeira camada deste sistema destinou-se a reconhecer gestos estáticos do conjunto de configurações de mãos identificados por Ferreira Brito (2010), apresentada na seção 3.2.1 desta dissertação, utilizando máquinas de vetores de suporte (SVMs) e redes neurais (ANNs). Já a segunda camada foi criada visando coletar as classificações originadas da primeira camada, anexar informações espaciais e de trajetória e então produzir uma classificação final da palavra sendo gesticulada, utilizando modelos ocultos de Markov (HMMs) e campos aleatórios condicionais ocultos (HCRFs).

Considerando o trabalho de Ferreira-Brito, para a elaboração deste sistema consideramos o vetor de características

$$\begin{gathered} \boldsymbol{x_t} = \langle h_c, h_{rx}, h_{ry}, h_\theta, f_\theta \rangle \\ \begin{aligned} h_c \in \mathbb{N}, \quad & 1 \leq h_c \leq 46 \\ h_x, h_y \in \mathbb{R} \quad & -1 \leq h_x, h_y \leq 1 \\ h_\theta, f_\theta \in \mathbb{R} \quad & -\pi \leq h_\theta, f_\theta \leq \pi \end{aligned} \end{gathered} \tag{8.2}$$

em que h_c denota a configuração de mão detectada pelo classificador de quadros individuais; h_{rx} e h_{ry} são as posições relativas da mão quando comparadas ao centro da face do usuário; e h_θ e f_θ são as orientações angulares das mãos e da face, respectivamente. Todas as posições relativas são normalizadas para o intervalo unitário, e toda informação angular será considerada em radianos.

Para maior clareza, nos referiremos ao conjunto das 46 configurações de mãos possíveis utilizando-se o símbolo $\mathbb{H}$. Para manter consistência com as seções subsequentes, o vetor de características será referido como $\boldsymbol{x_t}$, e cada elemento individual na posição i deste vetor será denotado $\boldsymbol{x_{t_i}}$.

8.2.1 Camada de reconhecimento de gestos estáticos

Continuando o fluxo de processamento, após as mãos terem sido localizadas, um banco de SVMs dispostas em um DDAG de máxima margem é utilizado para classificar a imagem das mãos em uma das 46 possíveis configurações de mãos de $\mathbb{H}$. Para tanto, os experimentos iniciais mostrados em (SOUZA, ANJO e PIZZOLATO, 2012) se mostraram particularmente uteis para aprender os modelos de aprendizado desta camada adequadamente, especialmente devido às heurísticas para seleção de hiperparâmetros exploradas anteriormente.

Os classificadores estáticos mostraram alguns resultados interessantes. A Tabela 7 mostra resultados selecionados das diversas configurações de SVMs testadas e exibe o número médio de avaliações de vetores de suporte (SV) como uma medida de esparsidade e eficiência. A Figura 75 exibe os valores de *Kappa* obtidos para diferentes escolhas da constante de complexidade C em uma máquina linear.

Tabela 7. Resultados para as máquinas classificadoras de configuração de mãos.

Função kernel	Estratégia de Decisão	Validação	Número de Vetores de Suporte	
		Kappa ± (0.95 C.I.)	*Total*	*Médio*
Linear	DDAG	0,2390 ± 0,0074	1035	45
	1-vs-1	0,2404 ± 0,0074		1.035
Quadrática	DDAG	0,4737 ± 0,0085	339509	8.069
	1-vs-1	0,4790 ± 0,0085		339.509
Gaussiana	DDAG	0,3401 ± 0,0042	375372	8.707
	1-vs-1	0,3417 ± 0,0042		375.372

[1] Média de avaliações únicas por vetor de suporte ao classificar o conjunto de instâncias de validação. No caso linear, este valor indica o número total de máquinas obtidas no problema de classificação em múltiplas classes.

A Figura 76 exibe o desempenho das ANNs em função do número de neurônios contidos em suas respectivas camadas ocultas. O melhor resultado reportado por uma ANN ocorreu utilizando-se 1000 neurônios na camada oculta com $\hat{\kappa}_{ANN} = 0{,}301$ e $\widehat{var}(\hat{\kappa}_{ANN}) = 4{,}05 \times 10^{-3}$, superior a melhor SVM linear, porém a um maior custo de processamento para seu aprendizado. É interessante notar que após este pico de desempenho, redes com maiores números de neurônios iniciaram a apresentação de *overfitting*.

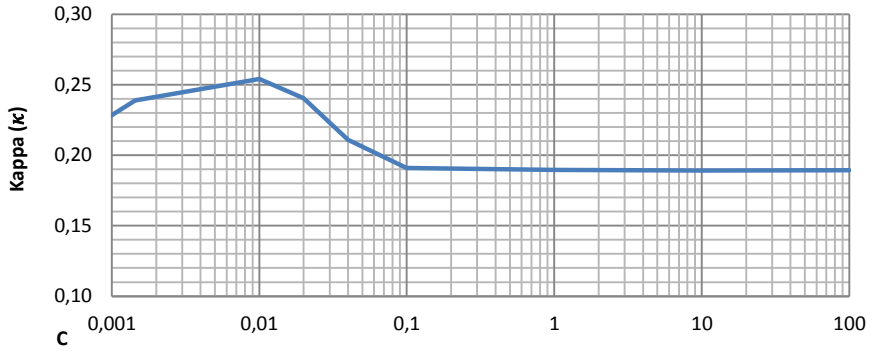

Figura 75. Resultados para as SVMs com função kernel linear em função do valor da constante de complexidade C no experimento com imagens de profundidade.

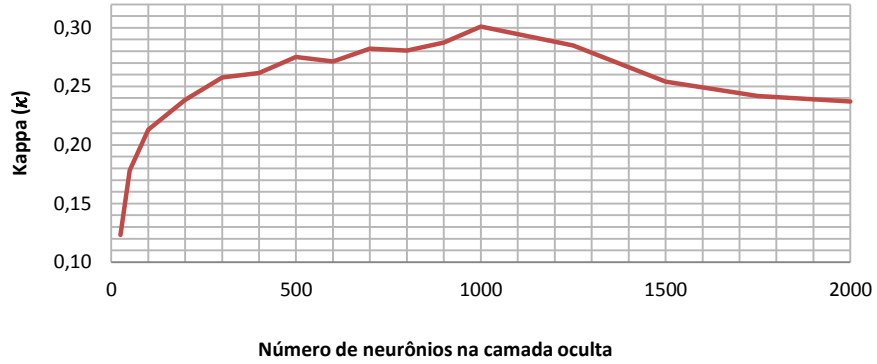

Figura 76. Resultados para as redes neurais classificadoras de configurações de mãos no experimento com imagens de profundidade.

Considerando isoladamente apenas a acurácia dos classificadores criados, uma DDAG-SVM quadrática seria a clara escolha para compor a primeira camada de classificação. Esta máquina apresentou resultados estatisticamente significantes e maiores que quaisquer outros modelos considerados neste experimento. No entanto, o custo computacional em se utilizar uma máquina quadrática ou mesmo Gaussiana pode ser demasiado grande, conforme evidenciado na Figura 77.

O uso de uma ANN neste caso também parece bastante convidativo. Por ter um tempo de avaliação constante, não requerem o mesmo custo computacional potencialmente variável das DDAG-SVMs quadráticas ou Gaussianas. Sua criação, em contrapartida, é bastante dispendiosa; pode requerer tempo e cuidados extras para seu aprendizado, principalmente considerando as múltiplas inicializações aleatórias para se alcançar um bom mínimo local.

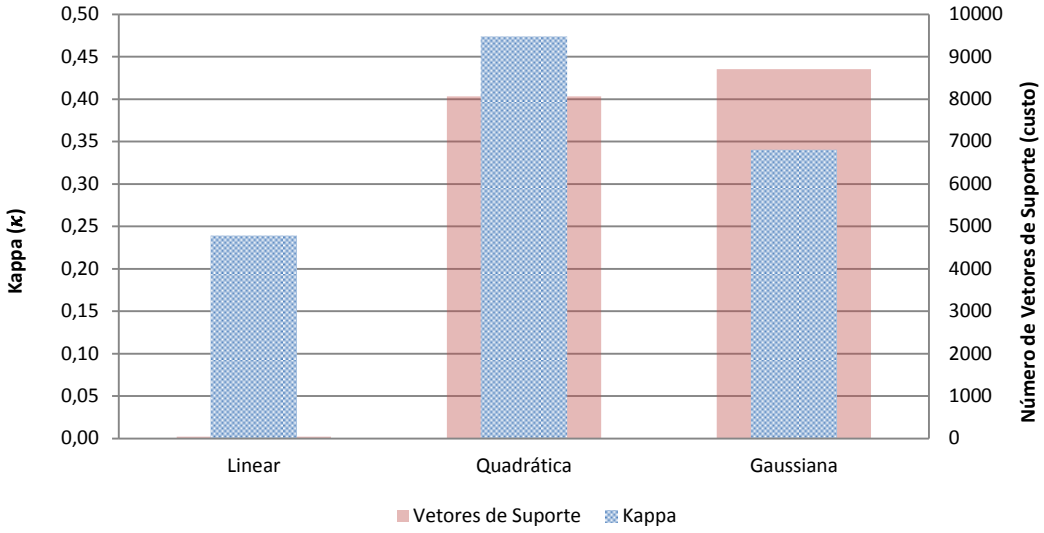

Figura 77. Gráfico de eficácia vs. eficiência para SVMs baseado nos valores de Kappa e número medio de avaliações de Vetores de Suporte no conjunto de dados avaliado.

Em contraste, o uso de uma DDAG-SVM novamente é capaz de unir ambas as qualidades de tempo de execução e de criação. Comparadas a máquinas quadráticas ou Gaussianas, o custo de se computar uma DDAG-SVM linear é bastante reduzido. Como um DDAG reduz o esforço computacional de computar 1035 decisões para apenas 45 decisões de tempo constante, novamente como no experimento com imagens em escala de cinza, a escolha de um DDAG linear proporciona um classificador bastante eficiente, porém relativamente fraco para o sistema de reconhecimento de gestos. Veremos na próxima seção que este desempenho reduzido não será problema devido à natureza probabilística dos modelos utilizados na segunda camada de processamento.

8.2.2 Camada de reconhecimento de gestos dinâmicos

O papel da segunda camada de processamento se inicia na obtenção da saída discreta da primeira camada, na combinação de informações espaciais, de trajetória e da face do usuário e na criação dos vetores de características exibidos na equação (8.2). Considerando cada vetor de características como uma observação individual $\boldsymbol{x_t}$ pertencente a uma sequência de observações $\boldsymbol{x}$, o objetivo esta camada de processamento é então estimar o rótulo de palavra ω mais provavelmente associado com um dado $\boldsymbol{x}$.

Para criar e aprender os modelos de gestos desta camada, consideramos funções de característica de ambas as naturezas discretas e continuas. Inicializamos nossos modelos HCRF com probabilidades extraídas de HMMs correspondentes, criados para operar sobre distribuições conjuntos mistas, de variáveis tanto discretas como continuas. Esta formulação de independência pode ser vista como a aplicação da hipótese de Bayes sobre nossos vetores de características.

Assim, as distribuições de emissão para as sequências de observações podem então ser expressas na forma

$$p(\boldsymbol{x_t}) = \prod_{i=1}^{5} p_i(\boldsymbol{x_{t_i}}) \tag{8.3}$$

em que $\boldsymbol{p_1}$ é uma distribuição discreta e $\boldsymbol{p_{2\ldots5}}$ são supostas aproximadamente normais, com parâmetros média e variância desconhecidos. Nota-se que pode haver certa preocupação quanto à validade destas suposições, já que distribuições Normais estariam sendo assumidas para variáveis de natureza circular (ou seja, cujos valores estão restritos ao intervalo $[-\pi; +\pi]$). No entanto, deve-se levar em consideração que a importância desta imprecisão pode ser diminuída uma vez que consideramos que tanto a face quanto as mãos do usuário têm sua liberdade de movimento restringida pelos pontos de junção do corpo humano, e, principalmente no caso da face, os valores destas variáveis não podem denotar uma volta completa.

A escolha de distribuições normais também facilita a expressão de nossos modelos. As características de um HCRF de cadeia linear podem ser denotadas

$$\begin{aligned}
f_{\omega'}^{(Label)}(\omega, \boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x_t}) &= \mathbf{1}_{\{\omega=\omega'\}} && \forall \omega' \in \Omega \\
f_{i,j}^{(Tr)}(\omega, \boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x_t}) &= \mathbf{1}_{\{y_t=i\}}\mathbf{1}_{\{y_{t-1}=j\}} && \forall i, j \in S_\omega \\
f_{i,o,d}^{(Em)}(\omega, \boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x_t}) &= \mathbf{1}_{\{y_t=i\}}\mathbf{1}_{\{(x_t)_d=o\}} && \begin{matrix}\forall i \in S_\omega \\ \forall o \in \mathbb{H} \\ \boldsymbol{x_{t_d}} \in \mathbb{N}\end{matrix} \\
f_{i,d}^{(Occ)}(\omega, \boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x_t}) &= \mathbf{1}_{\{y_t=i\}} && \begin{matrix}\forall i \in S_\omega \\ \boldsymbol{x_{t_d}} \in \mathbb{R}\end{matrix} \\
f_{i,d}^{(M1)}(\omega, \boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x_t}) &= \mathbf{1}_{\{y_t=i\}}(\boldsymbol{x_{t_d}}) && \begin{matrix}\forall i \in S_\omega \\ \boldsymbol{x_{t_d}} \in \mathbb{R}\end{matrix} \\
f_{i,d}^{(M2)}(\omega, \boldsymbol{y_t}, \boldsymbol{y_{t-1}}, \boldsymbol{x_t}) &= \mathbf{1}_{\{y_t=i\}}(\boldsymbol{x_{t_d}})^2 && \begin{matrix}\forall i \in S_\omega \\ \boldsymbol{x_{t_d}} \in \mathbb{R}\end{matrix}
\end{aligned} \tag{8.4}$$

em que Ω é o conjunto de todos possíveis rótulos de palavras em nosso problema de classificação, e S_ω é o número de estados assumido para sequências da classe ω. As características de rótulo $f_{\omega'}^{(Label)}$ acionam quando uma sequencia pertence à classe ω'. Características de transição $f_{i,j}^{(Tr)}$ acionam sempre que há uma transição do estado i para o estado j. Características de emissão $f_{i,o,d}^{(Em)}$ disparam quando um símbolo discreto o ocorre na posição d do vetor de observações enquanto o sistema está no estado i. Características de ocupação $f_{\mathrm{i,d}}^{(Occ)}$ acionam sempre que o estado i é alcançado, enquanto que as características de primeiro segundo memento, $f_{\mathrm{i,d}}^{(M1)}$ e $f_{i,d}^{(M2)}$ realizam a soma e a soma dos quadrados das características de observação na posição d quando o estado atual é i.

Utilizando este conjunto de funções de característica, um classificador baseado em HMMs criado sobre cada rótulo de classe ω com probabilidades *a priori* α_ω, matrizes de transição $\boldsymbol{A}_\omega$ e densidade de emissões $\boldsymbol{B}_\omega$ pode ser visto como um HCRF com seus respectivos componentes dados por

$$\begin{aligned}
\lambda_{\omega'}^{(Label)} &= \log \alpha_\omega && \forall \omega' \in \Omega \\
\lambda_{i,j}^{(Tr)} &= \log A_{i,j} && \forall i,j \in S_\omega \\
\lambda_{i,o,d}^{(Em)} &= \log B_i(o) && \begin{matrix}\forall i \in S_\omega \\ \forall o \in \mathbb{H} \\ \boldsymbol{x}_{t_d} \in \mathbb{N}\end{matrix} \\
\lambda_{i,d}^{(Occ)} &= -\frac{1}{2}\left(\log 2\pi\sigma^2 + \frac{{\mu_{i,d}}^2}{{\sigma_{i,d}}^2}\right) && \begin{matrix}\forall i \in S_\omega \\ \boldsymbol{x}_{t_d} \in \mathbb{R}\end{matrix} \\
\lambda_{i,d}^{(M1)} &= \frac{\mu_{i,d}}{{\sigma_{i,d}}^2} && \begin{matrix}\forall i \in S_\omega \\ \boldsymbol{x}_{t_d} \in \mathbb{R}\end{matrix} \\
\lambda_{i,d}^{(M2)} &= -\frac{1}{2{\sigma_{i,d}}^2} && \begin{matrix}\forall i \in S_\omega \\ \boldsymbol{x}_{t_d} \in \mathbb{R}\end{matrix}
\end{aligned} \tag{8.5}$$

em que $\mu_{i,d}$ e ${\sigma_{i,d}}^2$ se referem a media e a variância para a densidade de emissão do estado i para o elemento do vetor de observações na posição d caso este elemento tenha uma assumida distribuição Normal; e, no caso deste elemento ser discreto, $B_i(o)$ denota a função massa de probabilidade do estado i para o símbolo correspondente a uma configuração de mão $o \in \mathbb{H}$.

A Tabela 8 mostra os resultados obtidos para os modelos reconhecedores de sequências. A melhor combinação de modelos foi alcançada através da combinação dos modelos SVM e HCRF para compor a primeira se segunda camada de processamento, respectivamente.

Tabela 8. Resultados de classificação para os modelos de classificação de sequências (palavras).

Rotulação	Classificação	Treinamento	Validação
		Kappa ± (0.95 C.I.)	*Kappa ± (0.95 C.I.)*
SVM (quadrática)	HMM	0,8971 ± 0,0179	0,8071 ± 0,0222
	HCRF	**0,9218 ± 0,0156**	**0,8441 ± 0,0204**
SVM (linear)	HMM	0,8886 ± 0,0186	0,7967 ± 0,0227
	HCRF	0,9263 ± 0,0153	0,8198 ± 0,0217
ANN	HMM	0,8610 ± 0,0205	0,7473 ± 0,0771
	HCRF	0,9330 ± 0,0148	0,7727 ± 0,0743
Nenhum (somente trajetória)	*HMM*	*0,6411 ± 0,0287*	*0,5308 ± 0,0898*
	HCRF	*0,6638 ± 0,0283*	*0,5524 ± 0,0897*

Como podemos observar, houve uma significativa melhora nas taxas de reconhecimento ao utilizarmos SVMs e HCRFs. Em comparação ao uso de ANNs, a utilização de uma SVM com função kernel quadrática resultou em uma redução da taxa de erro em mais de 20% no caso de HMMs, e mais de 30% para HCRFs. No caso de uma SVM linear esta melhora foi mais discreta, de aproximadamente 19% para HMMs e 20% para HCRFs. Estas informações nos dão indícios de que o uso de SVM quadráticas na primeira camada de processamento leva a um melhor aproveitamento dos modelos gráficos discriminativos presentes na segunda camada.

Para confirmar esta hipótese, podemos verificar se esta diferença se mostra estatisticamente significante segundo um teste de *Kappa*. No caso das SVMs quadráticas, temos que

$$z = \frac{|\hat{\kappa}_{HMM} - \hat{\kappa}_{HCRF}|}{\sqrt{\widehat{Var}_0(\hat{\kappa}_{HMM}) + \widehat{Var}_0(\hat{\kappa}_{HCRF})}} = \frac{|0{,}8071 - 0{,}8441|}{\sqrt{0{,}01351^2 + 0{,}01240^2}} = 2{,}01473 \qquad (8.6)$$

o que nos dá um valor $p = 0{,}04393$, estatisticamente significante a um nível de significância $\alpha = 0{,}05$. No entanto, para as SVMs lineares temos

$$z = \frac{|\hat{\kappa}_{HMM} - \hat{\kappa}_{HCRF}|}{\sqrt{\widehat{Var}_0(\hat{\kappa}_{HMM}) + \widehat{Var}_0(\hat{\kappa}_{HCRF})}} = \frac{|0{,}7967 - 0{,}8197|}{\sqrt{0{,}01379^2 + 0{,}01316^2}} = 1{,}20818 \qquad (8.7)$$

e para as ANNs temos

$$z = \frac{|\hat{\kappa}_{HMM} - \hat{\kappa}_{HCRF}|}{\sqrt{\widehat{Var}_0(\hat{\kappa}_{HMM}) + \widehat{Var}_0(\hat{\kappa}_{HCRF})}} = \frac{|0{,}74932 - 0{,}7746|}{\sqrt{0{,}01491^2 + 0{,}01435^2}} = 1{,}22513 \quad (8.8)$$

o que não nos dá indícios suficientes para suportar a hipótese de que esta diferença seja significativa nestes dois casos ($p > 0{,}2$).

Desta análise, podemos concluir que a primeira etapa de processamento é crucial no melhor aproveitamento dos modelos da segunda camada. Quando alimentados com informações adequadas fornecidas através de uma SVM quadrática, os modelos discriminativos baseados em HCRFs apresentaram até 19% de ganho quando comparados a seus primos gerativos baseados em HMMs, diferença esta estatisticamente significante conforme apresentado na equação (8.6).

No entanto, aqui também apresentaremos uma visão alternativa sobre o que ocorre quando o classificador da primeira camada não é o melhor possível, como no caso de uma SVM com função kernel linear ou uma ANN. Nestes casos, podemos observar que, apesar da diferença de acurácia não ser estatisticamente significante para os conjuntos de validação, o mesmo não pode ser dito sobre os conjuntos de treinamento. Como no experimento com imagens em escala de cinza apresentado na seção 8.1, os modelos baseados em HCRFs não apresentaram *overfitting* – foi possível obter um ganho de desempenho estatisticamente significante ($p < 0.01$) sobre as amostras de treinamento sem ocasionar uma perda de generalidade sobre instâncias não observadas. Os modelos discriminativos se mostraram aptos a reter maior conhecimento sem perda de generalidade quando comparados a HMMs.

Finalmente, quando comparamos todas as variações de classificadores possíveis a modelos que não utilizaram informações sobre as configurações de mãos, isto é, a modelos que não utilizaram a primeira camada de processamento, podemos notar que os resultados são estatisticamente significantes para ambos os conjuntos de treinamento e teste por uma ampla margem.

Como indicado anteriormente, os relativamente baixos valores para κ reportados na Tabela 7 não foram empecilho para o desempenho geral do sistema. De fato, como em mecanismos de *boosting*, em que a combinação de classificadores fracos é capaz de gerar um classificador forte, aqui a segunda camada de processamento é capaz de detectar os padrões produzidos pela primeira camada, consolidando-os em informação notoriamente útil para a classificação de novos gestos. Assim e, portanto, a primeira camada de processamento efetivamente atua como um estágio de extração de características supervisionado, guiado pela informação linguística do

trabalho de Ferreira-Brito. Interessantemente, o acréscimo na capacidade de absorção de conhecimento relatada no parágrafo anterior também só foi observável na presença da primeira camada de processamento.

8.2.3 Conclusão deste experimento

Este experimento apresentou e consolidou a abordagem executada nesta dissertação para o reconhecimento de palavras em Libras. Através da combinação de SVMs dispostas em DDAGs aliadas a HCRFs, mostramos como o uso de modelos discriminativos, em contra partida a gerativos, auxiliou na melhora de desempenho do sistema apresentado sem que isso ocasionasse um provável *overfitting*. Mostramos como o uso de informação linguística auxiliou no projeto de tal sistema; e como a escolha de características simples, baseadas em um vetor de características misto com componentes discretos e contínuos se mostrou adequado nesta tarefa. Obtivemos resultados estatisticamente significantes favorecendo a escolha de modelos aqui defendida, e apresentamos uma visão alternativa caso preferíssemos favorecer a velocidade de reconhecimento ao invés de sua acurácia, como através do uso de SVMs com funções kernel lineares.

Aqui, pode-se notar que a primeira camada de processamento do sistema criado atua como um passo orientado e supervisionado de extração de características, ao invés de um estágio de processamento em si. O uso de uma camada de reconhecimento de configuração de mãos, baseada em DDAGs, se mostrou extremamente benéfica quando comparada a modelos que utilizaram somente informações espaciais e de trajetória, alcançando resultados estatisticamente significantes em comparação a modelos que não utilizaram esta técnica.

Capítulo 9

Conclusão e trabalhos futuros

"The best thing about the future is that it comes only one day at a time."
— Abraham Lincoln

Neste trabalho, apresentamos, desenvolvemos e conduzimos experimentos no problema de reconhecimento de gestos da Libras utilizando Máquinas de Vetores de Suporte e Campos Condicionais Aleatórios Ocultos. Na fundamentação deste trabalho, detalhamos o funcionamento destes modelos e a razão de sua utilização, bem como apresentamos as principais características da Libras e delimitamos o espaço de problemas enfrentados em sua caracterização. Em seu desenvolvimento, apresentamos e detalhamos nossa abordagem baseada em duas camadas de processamento para verificar a eficácia e eficiência destes modelos nos problemas enfrentados.

Nos experimentos realizados, identificamos que o desempenho de SVMs com funções *kernel* Gaussianas possui certa independência da escolha de hiperparâmetro C, estando seu desempenho melhor relacionado a uma escolha apropriada da constante σ^2. Também ressaltamos a importância de uma inicialização cautelosa dos pesos sinápticos nas ANNs através do uso de heurísticas de inicialização; bem como a dificuldade muitas vezes encontrada durante a tarefa de se encontrar um bom mínimo local para se cessar o treinamento. Encontramos nas SVMs lineares dispostas em DDAGs um bom compromisso entre eficácia e eficiência, tanto durante treinamento quanto execução.

Identificamos também que o desempenho desta primeira camada pode ser crucial em aumentar a capacidade de generalização do sistema de classificação como um todo. No entanto, também identificamos que, mesmo que este desempenho não seja ótimo – como é o caso quando preferimos as rápidas máquinas lineares – não há *perda* de generalidade na segunda etapa de processamento. Os HCRFs revelaram-se extremamente eficientes na retenção de conhecimento sem a deterioração do

aprendizado devido ao *overfitting*, quando comparados ao uso de HMMs. Finalmente, também determinamos o impacto da primeira camada de processamento testando sua influência em testes com ou não sua presença, revelando resultados estatisticamente significativos favorecendo a divisão do sistema – em diferentes camadas de classificação – utilizada neste trabalho.

Chegamos também à percepção de que a primeira camada de processamento serviu como uma etapa de extração de características supervisionada; que, guiada com as informações Linguísticas identificadas por Ferreira-Brito (2010), foi capaz de surtir resultados mensuráveis nas medidas de classificação finais de nosso sistema sem que para isto necessitasse exercer um resultado perfeito durante sua própria etapa de classificação intermediária.

Finalmente, devemos ressaltar que, em cada experimento realizado, encontramos pontos de partida para trabalhos futuros: estes se resumem, basicamente, ao teste de técnicas por hora não discutidas neste trabalho, como os Campos Aleatórios Latente-Dinâmicos, Máquinas de Vetores de Suporte Estruturais, e, talvez um dos mais interessantes a serem explorados, os modelos de Aprendizagem Profunda (*Deep Learning, DL*), cuja popularidade tem aumentado de maneira significativa e consistente nos últimos anos. Os resultados destes últimos têm se mostrado bastante promissores em áreas como o reconhecimento de voz e a tradução de voz simultânea em tempo real; o que nos leva a crer que estas técnicas também possam produzir resultados interessantíssimos quando aplicados ao problema de reconhecimento das Línguas de Sinais.

Assim, aqui no final deste trabalho, indicamos a trabalhos futuros que tenham esta dissertação como ponto de partida que explorem as técnicas disponíveis nessa nova área tão promissora.

Referências

Observando o mundo sobre o ombro de gigantes

ADVANCED MICRO DEVICES, INC. Advanced Micro Devices: AMD Extensions to the 3DNow!(TM) and MMX(TM) Instruction Sets - Manual (2000). **Advanced Micro Devices Web site**, March 2000. Disponivel em: <http://support.amd.com/us/Processor_TechDocs/22466.pdf>. Acesso em: 01 December 2011.

ALAHI, A.; ORTIZ, R.; VANDERGHEYNST, P. **FREAK:** Fast Retina Keypoint. Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. [S.l.]: IEEE. 2012. p. 510-517.

ALOSEFER, Y.; RANA, O. F. **Predicting client-side attacks via behaviour analysis using honeypot data**. Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on. [S.l.]: [s.n.]. 2011. p. 31-36.

ANJO, M. D. S. **Avaliação das Técnicas de Segmentação, Modelagem e Classificação para o Reconhecimento Automático de Gestos e Proposta de uma Solução para Classificar Gestos da Libras em Tempo Real**. Universidade Federal de São Carlos. São Carlos. 2012. Dissertação de Mestrado.

ANJO, M. D. S.; PIZZOLATO, E.; FEUERSTACK, S. **A Real-Time System to Recognize Static Hand Gestures of Brazilian Sign Language (Libras) alphabet using Kinect**. Proceedings of IHC 2012, the 6th Latin American Conference on Human-Computer Interaction. Cuiabá - Mato Grosso, Brazil: [s.n.]. 2012.

BAKER-SHENK, C. L. **A microanalysis of the nonmanual components of questions in American Sign Language**. Berkeley: University of California, 1983.

BALDI, P.; CHAUVIN, Y. Smooth On-line Learning Algorithms for Hidden Markov Models. **Neural Computation**, v. 6, n. 2, p. 307-318, 1994. Disponivel em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.41.3662>.

BANERJEE, M. et al. Beyond kappa: A review of interrater agreement measures. **Canadian Journal of Statistics**, v. 27, n. 1, p. 3-23, 1999. ISSN 1708-945X. Disponivel em: <http://dx.doi.org/10.2307/3315487>.

BARTLETT, M. S. et al. A prototype for automatic recognition of spontaneous facial actions. **Advances in Neural Information Processing Systems**, v. 15, p. 1271-1278, 2003.

BAUER, B.; KRAISS, K.-F. **Video-based sign recognition using self-organizing subunits**. Pattern Recognition, 2002. Proceedings. 16th International Conference on. [S.l.]: [s.n.]. 2002. p. 434-437.

BAUM, L. E. et al. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains. **The Annals of Mathematical Statistics**, v. 41, n. 1, 1970. Disponivel em: <http://dx.doi.org/10.2307/2239727>.

BAY, H. et al. Speeded-Up Robust Features (SURF). **Comput. Vis. Image Underst.**, New York, NY, USA, v. 110, n. 3, p. 346-359, June 2008. ISSN 1077-3142. Disponivel em: <http://dl.acm.org/citation.cfm?id=1370312.1370556>.

BERNARDES JÚNIOR, J. L. B. **Modelo abrangente e reconhecimento de gestos com as mãos livres para ambientes 3D**. Universidade de São Paulo (USP). São Paulo, Brazil. 2010.

BIRDWHISTELL, R. L. **Kinesics and Context**. [S.l.]: University of Pennsylvania Press, 1970.

BISHOP, C. M. **Pattern Recognition and Machine Learning (Information Science and Statistics)**. 1st ed. 2006. Corr. 2nd printing. ed. [S.l.]: Springer, 2007. Disponivel em: <http://www.worldcat.org/isbn/0387310738>.

BJÖRCK, Å. **Numerical methods for least squares problems**. [S.l.]: SIAM, 1996. ISBN 9780898713602.

BOBICK, A. F.; WILSON, A. D. A State-based Approach to the Representation and Recognition of Gesture. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 19, p. 1325-1337, 1997.

BOLT, R. A. "Put-that-there": Voice and gesture at the graphics interface. **SIGGRAPH Comput. Graph.**, New York, NY, USA, v. 14, n. 3, p. 262-270, July 1980. ISSN 0097-8930. Disponivel em: <http://doi.acm.org/10.1145/965105.807503>.

BOWDEN, R. et al. **A Linguistic Feature Vector for the Visual Interpretation of Sign Language**. European Conference on Computer Vision. [S.l.]: Springer-Verlag. 2004. p. 391-401.

BRADSKI, G. R. Computer Vision Face Tracking For Use in a Perceptual User Interface. **Intel Technology Journal**, n. Q2, 1998. Disponivel em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.14.7673>.

BRAFFORT, A. **ARGo:** An Architecture for Sign Language Recognition and Interpretation. Proceedings of Gesture Workshop on Progress in Gestural Interaction. London, UK: Springer-Verlag. 1997. p. 17-30.

BRASIL. Lei nº 10.436. Dispõe sobre a Língua Brasileira de Sinais - Libras e dá outras providências., 24 abril 2002.

BRETZNER, L.; LAPTEV, I.; LINDEBERG, T. **Hand Gesture Recognition using Multi-Scale Colour Features, Hierarchical Models and Particle Filtering**. Proc. Face and Gesture. [S.l.]: [s.n.]. 2002. p. 423-428.

BROWN, E. T. et al. **Dis-function:** Learning distance functions interactively. Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on. [S.l.]: [s.n.]. 2012. p. 83-92.

BRUMMITT, L. **Scrabble Referee: Word Recognition Component**. University of Sheffield. [S.l.]. 2011.

BRUNELLI, R.; POGGIO, T. Face Recognition: Features Versus Templates. **IEEE Trans. Pattern Anal. Mach. Intell.**, Washington, DC, USA, v. 15, n. 10, p. 1042-1052, October 1993. ISSN 0162-8828. Disponivel em: <http://dx.doi.org/10.1109/34.254061>.

BURSZTEIN, E.; MARTIN, M.; MITCHELL, J. C. **Text-based CAPTCHA strengths and weaknesses**. ACM Conference on Computer and Communications Security. [S.l.]: [s.n.]. 2011. p. 125-138.

CAPUTO, B. et al. **Appearance-based object recognition using SVMs:** which kernel should I use? Proceedings of the NIPS workshop on statistical methods for computational experiments in visual processing and computer vision. [S.l.]: Whistler. 2002.

CARNEIRO, A. T. S.; CORTEZ, P. C.; COSTA, R. C. S. **Reconhecimento de Gestos da LIBRAS com Classificadores Neurais a partir dos Momentos Invariantes de Hu**. Interaction 09 - South America. São Paulo: [s.n.]. 2009. p. 190-195.

CHEN, W. G. et al. **Transition movement models for large vocabulary continuous sign language recognition**. Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on. [S.l.]: [s.n.]. 2004. p. 553-558.

COHEN, J. A Coefficient of Agreement for Nominal Scales. **Educational and Psychological Measurement**, v. 20, n. 1, p. 37-46, apr 1960. Disponivel em: <http://dx.doi.org/10.1177/001316446002000104>.

COMANICIU, D.; RAMESH, V. **Robust Detection and Tracking of Human Faces with an Active Camera**. Proceedings of the Third IEEE International Workshop on Visual Surveillance (VS'2000). Washington, DC, USA: IEEE Computer Society. 2000.

CONGALTON, R. G. A review of assessing the accuracy of classifications of remotely sensed data. **Remote Sensing of Environment**, v. 37, n. 1, p. 35-46, 1991. Disponivel em: <http://linkinghub.elsevier.com/retrieve/pii/003442579190048B>.

COOLEY, J. W.; TUKEY, J. W. An Algorithm for the Machine Calculation of Complex Fourier Series. **Mathematics of Computation**, v. 19, n. 90, p. 297-301, Apr. 1965. Disponivel em: <http://www.jstor.org/stable/2003354>.

CORFIELD, D. et al. **Popper, Falsification and the VC-dimension**. Max Planck Institute for Biological Cybernetics. [S.l.]. 2005.

COVER, T. M. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. **IEEE Transactions on Electronic Computers**, v. EC-14, n. 3, p. 326 -334, June 1965. ISSN 0367-7508.

CRAMER, H. **Mathematical methods of statistics**. [S.l.]: Princeton University Press, 1946. ISBN 0691080046.

CRAMMER, K.; SINGER, Y. On the algorithmic implementation of multiclass kernel-based vector machines. **J. Mach. Learn. Res.**, v. 2, p. 265-292, March 2002. ISSN 1532-4435. Disponivel em: <http://dl.acm.org/citation.cfm?id=944790.944813>.

CRISTIANINI, N.; SHAWE-TAYLOR, J. **An introduction to support vector machines and other kernel-based learning methods**. 1. ed. Cambridge: Cambridge University Press, 2000. ISBN 0521780195.

CROW, F. C. **Summed-area tables for texture mapping**. SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques. New York, NY, USA: ACM. 1984. p. 207-212.

CYBENKO, G. Approximations by superpositions of sigmoidal functions. **Mathematics of Control, Signals, and Systems (MCSS)**, New York, v. 2, p. 303-314, 1989. Disponivel em: <http://dx.doi.org/10.1007/BF02551274>.

DAHL, G. E. et al. **Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition**. IEEE Transactions on Audio, Speech, and Language Processing. [S.l.]: [s.n.]. 2012.

DALAL, N.; TRIGGS, B. **Histograms of oriented gradients for human detection**. Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. [S.l.]: [s.n.]. 2005. p. 886-893.

DAVIS, J.; SHAH, M. **Visual gesture recognition**. Vision, Image and Signal Processing, IEEE Proceedings on. [S.l.]: [s.n.]. April 1994. p. 101 -106.

DIAS, D. B. et al. **Hand movement recognition for brazilian sign language:** a study using distance-based neural networks. Proceedings of the 2009 international joint conference on Neural Networks. Atlanta, Georgia, USA: IEEE Press. 2009. p. 2355-2362.

DIETTERICH, T. G.; BAKIRI, G. Solving multiclass learning problems via error-correcting output codes. **Journal of Artificial Intelligence Research**, v. 2, p. 263-286, 1995.

DUDA, R. O.; MACHANIK, J. W. **Function modeling experiments**. Stanford Research. [S.l.]. 1963.

EKMAN, P. Facial Expression and Emotion. **American Psychologist**, v. 48, n. 4, p. 384-392, 1993.

ELMEZAIN, M. O. S. M. **Hand gesture spotting and recognition using HMMs and CRFs in color image sequences**. Magdeburg, University. [S.l.]. 2010.

ELMEZAIN, M.; AL-HAMADI, A.; MICHAELIS, B. **A Novel System for Automatic Hand Gesture Spotting and Recognition in StereoColor Image Sequences**. 17th International Conference on Computer Graphics, Visualization and Computer Vision (WSCG 2009). Bory, Czech Republic: Springer. 2009. p. 355-359.

ELMEZAIN, M.; AL-HAMADI, A.; MICHAELIS, B. **Hand trajectory-based gesture spotting and recognition using HMM**. Proceedings of the International Conference on Image Processing, ICIP 2009. Cairo, Egypt: IEEE. 2009. p. 3577-3580.

ELMEZAIN, M.; AL-HAMADI, A.; MICHAELIS, B. **A Robust Method for Hand Gesture Segmentation and Recognition**. 20th International Conference on Pattern Recognition, ICPR. Istanbul, Turkey: IEEE. 2010. p. 3850-3853.

EUGENIO, B. D. **On the usage of Kappa to evaluate agreement on coding tasks**. In Proceedings of the Second International Conference on Language Resources and Evaluation. [S.l.]: [s.n.]. 2000. p. 441-444.

EVANS, C. **Notes on the OpenSURF Library**. University of Bristol. [S.l.]. 2009. (CSTR-09-001).

FAMULARO, R. Intervención del interprete de senas / lengua oral en el contrato pedagógico de la integración. In: SKLIAR, C. **Atualidades da educação bilíngüe para surdos**. 2nd. ed. Porto Alegre: Mediação, v. 1, 1999. p. 272. ISBN 85-87063-26-X.

FELTY, W. L.; JURS, P. C. Multicategory prediction using arrays of binary pattern classifiers. **Analytical Chemistry**, v. 45, n. 6, p. 885-889, 1973. Disponivel em: <http://pubs.acs.org/doi/abs/10.1021/ac60328a012>.

FERREIRA-BRITO, L. **Por uma gramática de Línguas de Sinais**. 2nd. ed. Rio de Janeiro: Tempo Brasileiro, 2010. 273 p. ISBN 85-282-0069-8.

FERREIRA-BRITO, L.; LANGEVIN, R. The Sublexical Structure of a Sign Language. **Mathématiques, Informatique et Sciences Humaines**, v. 125, p. 17-40, 1994.

FERRI, C.; HERNÁNDEZ-ORALLO, J.; MODROIU, R. An experimental comparison of performance measures for classification. **Pattern Recogn. Lett.**, New York, NY, USA, v. 30, n. 1, p. 27-38, January 2009. ISSN 0167-8655. Disponivel em: <http://dl.acm.org/citation.cfm?id=1458730.1458898>.

FEUERSTACK, S.; ANJO, M. D. S.; PIZZOLATO, E. **Model-based Design, Generation and Evaluation of a Gesture-based User Interface Navigation Control**. Proceedings of the 5th Latin American Conference on Human-Computer Interaction (IHC 2011). Porto de Galinhas, Brazil: [s.n.]. 2011.

FEUERSTACK, S.; COLNAGO, J. H.; SOUZA, C. R. D. **Designing and Executing Multimodal Interfaces for the Web based on State Chart XML**. Proceedings of 3a. Conferência Web W3C Brasil 2011. Rio de Janeiro: [s.n.]. 2011.

FISHER, R. A. On the Mathematical Foundations of Theoretical Statistics. **Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character**, p. 309-368, 1922.

FLEISS, J. L. Measuring nominal scale agreement among many raters. **Psychological Bulletin**, v. 76, n. 5, p. 378-382, 1971. ISSN 0033-2909.

FLEISS, J. L.; COHEN, J.; EVERITT, B. S. Large sample standard errors of kappa and weighted kappa. **Psychological Bulletin**, n. 5, p. 323-327, 1969. ISSN 0033-2909.

FLEISS, J. L.; LEVIN, B. A.; PAIK, M. C. **Statistical methods for rates and proportions**. 3rd. ed. [S.l.]: J. Wiley, 2003. ISBN 9780471526292.

FRANC, V.; HLAVAC, V. **Multi-class support vector machine**. Pattern Recognition, 2002. Proceedings of the 16th International Conference on. [S.l.]: [s.n.]. 2002. p. 236-239.

FREEDMAN, B. et al. **Depth Mapping using Projected Patterns**. US 2008/0240502 A1, 6 Sep 2007.

FRIEDMAN, J. H. **Another approach to polychotomous classification**. Stanford University. [S.l.]. 1996.

FUJIE, S. et al. **A conversation robot using head gesture recognition as para-linguistic information**. Robot and Human Interactive Communication, 2004. ROMAN 2004. 13th IEEE International Workshop on. [S.l.]: [s.n.]. 2004. p. 159-164.

FUKUNAGA, K.; HOSTETLER, L. The estimation of the gradient of a density function, with applications in pattern recognition. **Information Theory, IEEE Transactions on**, v. 21, p. 32-40, 1975. ISSN 0018-9448. Disponivel em: <http://dx.doi.org/10.1109/TIT.1975.1055330>.

GAO, W. et al. **Transition movement models for large vocabulary continuous sign language recognition**. Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on. [S.l.]: [s.n.]. 2004. p. 553-558.

GENTON, M. G. Classes of kernels for machine learning: a statistics perspective. **J. Mach. Learn. Res.**, v. 2, p. 299-312, March 2002. ISSN 1532-4435. Disponivel em: <http://dl.acm.org/citation.cfm?id=944790.944815>.

GHOBADI, S. et al. **Hand Segmentation Using 2D/3D Images**. IVCNZ 2007 Conference. Hamilton, New Zealand: University of Waikato. 2007. p. 64-69.

GROBEL, K.; ASSAN, M. **Isolated sign language recognition using hidden Markov models**. Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on. [S.l.]: [s.n.]. 1997. p. 162-167.

GUARINELLO, A. C. et al. O intérprete universitário da Língua Brasileira de Sinais na cidade de Curitiba. **Revista Brasileira de Educação Especial**, Marília, v. 14, n. 1, p. 63-74, Abril 2008. ISSN ISSN 1413-6538. Disponivel em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1413-65382008000100006&lng=en&nrm=iso>.

GUNAWARDANA, A. et al. **Hidden conditional random fields for phone classification**. Proceedings of Interspeech'2005. [S.l.]: [s.n.]. 2005. p. 1117-1120.

HAGAN, M. T.; MENHAJ, M. B. Training feedforward networks with the Marquardt algorithm. **IEEE Transactions on Neural Networks**, v. 5, p. 989-993, 6 November 1994. ISSN 10459227. Disponivel em: <http://dx.doi.org/10.1109/72.329697>.

HASSANI, A. Z. **Touch versus in-air Hand Gestures: Evaluating the acceptance by seniors of Human-Robot Interaction using Microsoft Kinect**. University of Twente. Enschede, The Netherlands. 2011.

HASSANIEH, H. et al. Nearly Optimal Sparse Fourier Transform. **CoRR**, v. abs/1201.2501, 2012. Disponivel em: <http://arxiv.org/abs/1201.2501>.

HASSOUN, M. H. **Fundamentals of Artificial Neural Networks**. 1st. ed. Cambridge: MIT Press, 1995. ISBN 026208239X.

HASTIE, T. et al. The Entire Regularization Path for the Support Vector Machine. **J. Mach. Learn. Res.**, v. 5, p. 1391-1415, December 2004. ISSN 1532-4435. Disponivel em: <http://dl.acm.org/citation.cfm?id=1005332.1044706>.

HELMER, S.; LOWE, D. **Using stereo for object recognition**. Robotics and Automation (ICRA), 2010 IEEE International Conference on. [S.l.]: [s.n.]. 2010. p. 3121-3127.

HOLDEN, E.-J.; LEE, G.; OWENS, R. Australian sign language recognition. **Machine Vision and Applications**, v. 16, n. 5, p. 312-320, 2005. ISSN 0932-8092. Disponivel em: <http://dx.doi.org/10.1007/s00138-005-0003-1>.

HONG, P.; HUANG, T. S.; TURK, M. Gesture Modeling and Recognition Using Finite State Machines. **Automatic Face and Gesture Recognition, IEEE International Conference on**, Los Alamitos, CA, USA, 2000. ISSN 0-7695-0580-5. Disponivel em: <http://dx.doi.org/10.1109/AFGR.2000.840667>.

HORNIK, K. Approximation capabilities of multilayer feedforward networks. **Neural Networks**, v. 4, n. 2, p. 251-257, 1991. Disponivel em: <http://linkinghub.elsevier.com/retrieve/pii/089360809190009T>.

HU, M.-K. Visual pattern recognition by moment invariants. **Information Theory, IRE Transactions on**, v. 8, February 1962.

HUBEL, D. H.; WIESEL, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. **The Journal of physiology**, v. 160, p. 106-154, 1962. Disponivel em: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1359523/>.

IBGE. **Censo Demográfico 2000:** Características gerais da população - Resultados da amostra. Rio de Janeiro: Instituto Brasileiro de Geografia e Estatística - IBGE, 2003. 1-178 p. ISBN 0104-3145. Disponivel em: <http://www.ibge.gov.br/home/estatistica/populacao/censo2000/populacao/censo2000_populacao.pdf>. Acesso em: 10 jan. 2012.

IBGE. **Censo Demográfico 2010 - Características Gerais da População, Religião e Pessoas com Deficiência**. Rio de Janeiro: Instituto Brasileiro de Geografia e Estatística - IBGE, 2010. 1-215 p. Disponivel em: <ftp://ftp.ibge.gov.br/Censos/Censo_Demografico_2010/Caracteristicas_Gerais_Religiao_Deficiencia/caracteristicas_religiao_deficiencia.pdf>. Acesso em: 26 dez. 2012.

IGEL, C.; HÜSKEN, M. **Improving the Rprop Learning Algorithm**. Symposium A Quarterly Journal In Modern Foreign Literatures. [S.l.]: [s.n.]. 2000. p. 115-121.

INTEL CORPORATION. **IA-32 Intel Architecture Software Developer's Manual Volume 2:** Instruction Set Reference. [S.l.]: [s.n.], 2003.

ISHIBUCHI, K.; TAKEMURA, H.; KISHINO, F. **Real time hand gesture recognition using 3D prediction model**. Systems, Man and Cybernetics, 1993. 'Systems Engineering in the Service of Humans', International Conference on. [S.l.]: [s.n.]. 1993. p. 324-328.

JAAKKOLA, T.; DIEKHANS, M.; HAUSSLER, D. **Using the Fisher Kernel Method to Detect Remote Protein Homologies**. [S.l.]: AAAI Press. 1999. p. 149-158.

JOACHIMS, T. Text categorization with Support Vector Machines: Learning with many relevant features Machine Learning. In: NÉDELLEC, C.; ROUVEIROL, C. **Machine Learning:** ECML-98. Berlin/Heidelberg: Springer Berlin / Heidelberg, v. 1398, 1998. Cap. 19, p. 137-142. ISBN 978-3-540-64417-0. Disponivel em: <http://dx.doi.org/10.1007/BFb0026683>. Lecture Notes in Computer Science.

JOACHIMS, T. A support vector method for multivariate performance measures. **Proceedings of the 22nd International Conference on Machine Learning**, p. 377-384, 2005.

JOHNSON, R. E.; FOOTE, B. **Designing Reusable Classes**. Domain Analysis and Software Systems Modeling. Los Alamitos, CA, USA: IEEE Computer Society Press. Também publicado em Junho/Julho de 1988 no Journal of Object-Oriented Programming (JOOP), mas este jornal parece não estar mais em impressão ou facilmente disponível.

JUANG, B. H.; RABINER, L. R. Automatic Speech Recognition--A Brief History of the Technology. In: (ORG.), K. B. **Elsevier Encyclopedia of Language and Linguistics**. 2nd. ed. [S.l.]: Oxford: Elsevier, 2005.

JURIE, F.; DHOME, M. **Real Time Robust Template Matching**. in British Machine Vision Conference 2002. [S.l.]: [s.n.]. 2002. p. 123-131.

KAPOOR, A.; PICARD, R. W. **A real-time head nod and shake detector**. Proceedings of the 2001 workshop on Perceptive user interfaces. Orlando, Florida: ACM. 2001. p. 1-5.

KEERTHI, S. S. et al. Improvements to Platt's SMO Algorithm for SVM Classifier Design. **Neural Comput.**, Cambridge, MA, USA, v. 13, n. 3, p. 637-649, March 2001. ISSN 0899-7667. Disponivel em: <http://dl.acm.org/citation.cfm?id=1120489.1120499>.

KENDALL, M. G. A New Measure of Rank Correlation. **Biometrika**, v. 30, p. 81-93, 1938. ISSN 00063444.

KENDON, A. **Gesture:** visible action as utterance. [S.l.]: Cambridge University Press, 2004. ISBN 9780521542937.

KHOSHELHAM, K. **Accuracy analysis of kinect depth data**. International Archives of Photogrammetry and Remote Sensing : IAPRS : ISPRS ; XXXVIII-5/W12. Calgary, Canada: International Society for Photogrammetry and Remote Sensing (ISPRS). 2011. p. 6.

KINDERMANN, R.; SNELL, J. L. **Markov Random Fields and Their Applications**. [S.l.]: American Mathematical Society, 1980. ISBN 9780821850015.

KIRILLOV, A. The AForge.NET Framework. **AForge.NET**, 2012. Disponivel em: <http://www.aforgenet.com/framework/>. Acesso em: 2 jan. 2012.

KNERR, S.; PERSONNAZ, L.; DREYFUSS, G. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In: FOGELMANN, J. **Neurocomputing:** Algorithm, Architectures and Applications. [S.l.]: Springer, 1990.

KRIPPENDORFF, K. H. **Content Analysis; An Introduction to its Methodology**. Beverly Hills, CA: Sage Publications, Inc, 1980. 188 p.

LAFFERTY, J. D.; MCCALLUM, A.; PEREIRA, F. C. N. **Conditional Random Fields:** Probabilistic Models for Segmenting and Labeling Sequence Data. Proceedings of the Eighteenth International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2001. p. 282-289.

LANDIS, J. R.; KOCH, G. G. The measurement of observer agreement for categorical data, v. 33, n. 1, 1977. Disponivel em: <http://www.ncbi.nlm.nih.gov/pubmed/843571>.

LECUN, Y. et al. Learning Algorithms For Classification: A Comparison On Handwritten Digit Recognition. In: _____ **Neural Networks:** The Statistical Mechanics Perspective. [S.l.]: World Scientific, 1995. p. 261-276.

LEE, H.-K.; KIM, J. H. An HMM-Based Threshold Model Approach for Gesture Recognition. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, p. 961-973, 1999.

LEE, Y.; LIN, Y.; WAHBA, G. Multicategory Support Vector Machines, theory, and application to the classification of microarray data and satellite radiance data. **Journal of the American Statistical Association**, v. 99, p. 67-81, 2004.

LEVADA, A. L. M. **Combinação de modelos de campos aleatórios markovianos para classificação contextual de imagens multiespectrais**. Universidade de São Paulo. São Carlos. 2010. Tese (Doutorado em Física Aplicada).

LEWIS, J. P. Fast Template Matching Template. **Pattern Recognition**, v. 10, n. 11, p. 120-123, 1995. Disponivel em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.157.3888&rep=rep1&type=pdf>.

LEWIS, M. P. (Ed.). **Ethnologue:** Languages of the World. 16th. ed. Dallas: SIL International, 2009. Disponivel em: <http://www.ethnologue.com/>. Acesso em: 08 August 2011.

LIDEGAARD, M. **Development of a Head Mounted Device for Point-of-Gaze Estimation in Three Dimensions**. The Maersk Mc-Kinney Moller Institute, University of Southern Denmark. [S.l.]. 2012. Tese de Mestrado.

LIENHART, R.; MAYDT, J. **An extended set of Haar-like features for rapid object detection**. Proceeding of the International Conference on Image Processing 2002. [S.l.]: IEEE. 2002. p. 900-903.

LIN, H.-T.; LIN, C.-J.; WENG, R. C. A note on Platt's probabilistic outputs for support vector machines. **Mach. Learn.**, Hingham, MA, USA, v. 68, n. 3, p. 267-276, October 2007. ISSN 0885-6125. Disponivel em: <http://dl.acm.org/citation.cfm?id=1286062.1286078>.

LIN, Y. Support vector machines and the Bayes rule in classification. **Data Mining and Knowledge Discovery**, v. 6, p. 259-275, 2002.

LIN, Y.; LEE, Y.; WAHBA, G. Support Vector Machines for Classification in Nonstandard Situations. **Mach. Learn.**, Hingham, MA, USA, v. 46, n. 1-3, p. 191-202, March 2002. ISSN 0885-6125. Disponivel em: <http://dx.doi.org/10.1023/A:1012406528296>.

LIRA, G. D. A.; SOUZA, T. A. F. D. Dicionário da Língua Portuguesa de Sinais. **Acessibilidade Brasil**, 2008. Disponivel em: <http://www.acessobrasil.org.br/libras/>. Acesso em: 07 nov. 2011.

LIU, C.-L. et al. Handwritten digit recognition: benchmarking of state-of-the-art techniques. **Pattern Recognition**, v. 36, n. 10, p. 2271-2285, 2003. ISSN 0031-3203. Disponivel em: <http://www.sciencedirect.com/science/article/pii/S0031320303000852>.

LONG, J. S. **The sign language:** a manual of signs. [S.l.]: Press of Gibson bros., 1910. Disponivel em: <http://www.archive.org/details/signlanguagemanu00longuoft>.

LOURENÇO, J. **Wii3D: Extending the Nintendo Wii Remote into 3D**. Rhodes University. Grahamstown, South Africa. 2010.

LUCAS, C. **The sociolinguistics of sign languages**. [S.l.]: Cambridge University Press, 2001. ISBN 9780521794749.

MACCLELLAND, J. L.; RUMELHART, D. E. **Parallel distributed processing:** explorations in the microstructure of cognition, vol. 1: foundations. Cambridge: MIT Press, v. 1, 1986. ISBN 0-262-68053-X.

MACKAY, D. J. C. A practical Bayesian framework for backpropagation networks. **Neural Computation**, Cambridge, MA, USA, v. 4, n. 3, p. 448-472, may 1992. ISSN 0899-7667. Disponivel em: <http://citeseer.ifi.unizh.ch/mackay91practical.html>.

MAHAJAN, M.; GUNAWARDANA, A.; ACERO, A. **Training algorithms for hidden conditional random fields**. Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP). [S.l.]: [s.n.]. 2006. p. 273-276.

MARQUARDT, D. W. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. **SIAM Journal on Applied Mathematics**, v. 11, n. 2, p. 431-441, 1963. Disponivel em: <http://link.aip.org/link/?SMM/11/431/1>.

MCCULLOCH, W.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **Bulletin of Mathematical Biology**, v. 5, n. 4, p. 115-133, 1943. ISSN 0092-8240. Disponivel em: <http://dx.doi.org/10.1007/BF02478259>.

MENDELSSOHN, T. **GestureBoard - Entwicklung eines Wiimote-basierten, gestengesteuerten Whiteboard-Systems für den Bildungsbereich**. University of Furtwangen. [S.l.]. 2010.

MERCER, J. Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations. **Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character**, London, v. 209, n. 441-458, p. 415-446, 1909. Disponivel em: <http://rsta.royalsocietypublishing.org/content/209/441-458/415.short>.

MICHEL, P.; EL KALIOUBY, R. **Real time facial expression recognition in video using support vector machines**. Proceedings of the 5th international conference on Multimodal interfaces. Vancouver, British Columbia, Canada: ACM. 2003. p. 258-264.

MINKA, T. **Discriminative models, not discriminative training**. Microsoft Research. [S.l.]. 2005.

MINSKY, L. M.; PAPERT, A. S. **Perceptrons**. Cambridge, MA: MIT Press, 1969.

MINSKY, M. **Steps toward artificial intelligence**. Proceedings of the Institute of Radio Engineers. [S.l.]: [s.n.]. 1961. p. 8-30. Reprinted in E. A. Feigenbaum and J. Feldman, editors, Computers and Thought, McGraw-Hill, New York, pp. 406-450, 1963.

MITRA, S.; ACHARYA, T. Gesture recognition: A survey. **IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews**, v. 37, n. 3, p. 311-324, 2007.

MOHAN, A.; PAPAGEORGIOU, C.; POGGIO, T. Example based object detection in images by components. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 23, p. 349-361, 2001.

MONTANA, D. J.; DAVIS, L. **Training feedforward neural networks using genetic algorithms**. Proceedings of the 11th international joint conference on Artificial intelligence - Volume 1. Detroit, Michigan: Morgan Kaufmann Publishers Inc. 1989. p. 762-767.

MURPHY, K. P.; WEISS, Y.; JORDAN, M. I. **Loopy Belief Propagation for Approximate Inference:** An Empirical Study. In Proceedings of Uncertainty in AI. [S.l.]: [s.n.]. 1999. p. 467-475.

NAGY, G. State of the art in pattern recognition. **Proceedings of the IEEE**, v. 56, n. 5, p. 836-863, may 1968. ISSN 0018-9219.

NG, A. Y.; JORDAN, M. I. On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes. In: DIETTERICH, T. G.; BECKER, S.; GHAHRAMANI, Z. **Neural Information Processing Systems**. [S.l.]: MIT Press, v. 2, 2001. p. 841-848.

NGUYEN, D.; WIDROW, B. **Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights**. Neural Networks, 1990., 1990 IJCNN International Joint Conference on. [S.l.]: [s.n.]. 1990. p. 21-26.

NGUYEN, T. D.; RANGANATH, S. **Towards recognition of facial expressions in sign language:** Tracking facial features under occlusion. Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on. [S.l.]: [s.n.]. 2008. p. 3228-3231.

NICOLOSO, S.; SILVA, S. M. D. Lendo sinalizações em Libras: onde está o sujeito? In: QUADROS, R. M. D.; MARIANE, R. S. **Estudos Surdos IV**. Petrópolis, RJ: Arara Azul, 2009.

O'BRIEN, J. **The production of reality:** essays and readings on social interaction. [S.l.]: Pine Forge Press, 2005.

OIKONOMIDIS, I.; KYRIAZIS, N.; ARGYROS, A. **Efficient model-based 3D tracking of hand articulations using Kinect**. Proceedings of the British Machine Vision Conference. [S.l.]: BMVA Press. 2011.

OSUNA, E.; FREUND, R.; GIROSIT, F. **Training support vector machines:** an application to face detection. Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on. [S.l.]: [s.n.]. 1997. p. 130-136.

OTSU, N. A threshold selection method from gray-level histograms. **IEEE Transactions on Systems, Man and Cybernetics**, v. 9, n. 1, p. 62-66, jan 1979.

PAPAGEORGIOU, C. P.; OREN, M.; POGGIO, T. A general framework for object detection. **Sixth International Conference on Computer Vision**, v. 6, p. 555-562, January 1998. Disponivel em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=710772>.

PEARL, J. **Reverend Bayes on inference engines:** A distributed hierarchical approach. Proceedings of the American Association of Artificial Intelligence National Conference on AI. Pittsburgh, PA: [s.n.]. 1982. p. 133-136.

PEARSON, K. **On the theory of contingency and its relation to association and normal correlation**. [S.l.]: Dulau and co., 1904.

PENG, F.; FENG, F.; MCCALLUM, A. **Chinese segmentation and new word detection using conditional random fields**. Proceedings of the 20th international conference on Computational Linguistics. Geneva, Switzerland: Association for Computational Linguistics. 2004.

PENG, F.; MCCALLUM, A. **Accurate information extraction from research papers using conditional random fields**. Proceedings of Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics (HLT-NAACL). [S.l.]: [s.n.]. 2004. p. 329-336.

PIZZOLATO, E. B.; ANJO, M. D. S.; PEDROSO, G. C. **Automatic recognition of finger spelling for LIBRAS based on a two-layer architecture**. Proceedings of the 2010 ACM Symposium on Applied Computing. Sierre, Switzerland: ACM. 2010. p. 969-973.

PLAGEMANN, C. et al. **Real-time identification and localization of body parts from depth images**. Robotics and Automation (ICRA), 2010 IEEE International Conference on. [S.l.]: [s.n.]. 2010. p. 3108 -3113.

PLATT, J. C. Sequential minimal optimization: A fast algorithm for training support vector machines. **Advances in Kernel Methods and Support Vector Learning**, v. 208, p. 1-21, 1998. Disponivel em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.55.560&rep=rep1&type=pdf>.

PLATT, J. C. Fast training of support vector machines using sequential minimal optimization. In: SCHÖLKOPF, B.; BURGES, C. J. C.; SMOLA, A. J. **Advances in kernel methods**. Cambridge, MA, USA: MIT Press, 1999. p. 185-208. ISBN 0-262-19416-3. Disponivel em: <http://dl.acm.org/citation.cfm?id=299094.299105>.

PLATT, J. C. **Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods**. Advances in Large Margin Classifiers. [S.l.]: MIT Press. 1999. p. 61-74.

PLATT, J. C.; CRISTIANINI, N.; SHAWE-TAYLOR, J. **Large Margin DAGs for Multiclass Classification**. Advances in Neural Information Processing Systems. [S.l.]: MIT Press. 2000. p. 547-553.

POLAND, J. **On the Robustness of Update Strategies for the Bayesian Hyperparameter alpha**. [S.l.]. 2001.

POPPER, K. R. **The Logic of Scientific Discovery**. London: Hutchinson, 1959. 479 p.

PREE, W.; KOSKIMIES, K. **Framelets - Small is Beautiful**. [S.l.]: [s.n.]. 1999.

QUADROS, R. M. D. Phrase Structure of Brazilian Sign Language. In: BAKER, A.; BOGAERDE, B. V. D.; CRASBORN., O. **Cross-linguistic perspectives in sign language research**. 1 ed. ed. Hamburg: Signum, v. 41, 2003. p. 141-162.

QUADROS, R. M.; KARNOPP, L. B. **Língua de sinais brasileira:** estudos linguísticos. [S.l.]: Artmed Editora, 2004. ISBN 9788536303086.

QUATTONI, A.; COLLINS, M.; DARRELL, T. **Conditional Random Fields for Object Recognition**. Advances in Neural Information Processing Systems 17. Cambridge, MA: MIT Press. 2005. p. 1097-1104.

RABINER, L. R. A tutorial on hidden Markov models and selected applications in speech recognition. In: WAIBEL, A.; LEE, K.-F. **Readings in speech recognition**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990. p. 267-296. ISBN 1-55860-124-4. Disponivel em: <http://dl.acm.org/citation.cfm?id=108235.108253>.

RABINER, L. R.; JUANG, B. H. **Fundamentals of speech recognition**. [S.l.]: PTR Prentice Hall, 1993. ISBN 9780130151575.

RIEDMILLER, M. **RProp - Description and Implementation Details**. University of Karlsruhe. Karlsruhe. 1994.

ROSA, A. D. S. A presença do intérprete de língua de sinais na mediação social entre surdos e ouvintes. In: SILVA, I. R.; KAUCHAKJE, S.; GESUELI, Z. M. **Cidadania, surdez e linguagem:** desafios e realidades. [S.l.]: Plexus, 2003.

ROSA, A. M. **(Res) Significando a Questão da Linguagem no Trabalho com a Criança Surda**. Pontifícia Universidade Católica de São Paulo, PUC/SP. São Paulo, Brasil, p. 156. 2008.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. **Psychological Review**, v. 65, n. 6, p. 386-408, November 1958.

ROWLEY, H. A.; BALUJA, S.; KANADE, T. Neural Network-Based Face Detection. **IEEE Transactions On Pattern Analysis and Machine intelligence**, v. 20, p. 23-38, 1998.

SAARINEN, S.; BRAMLEY, R.; CYBENKO, G. Ill-conditioning in neural network training problems. **SIAM J. Sci. Comput.**, Philadelphia, PA, USA, v. 14, n. 3, p. 693-714, May 1993. ISSN 1064-8275. Disponivel em: <http://dx.doi.org/10.1137/0914044>.

SALVI, J.; PAGÈS, J.; BATLLE, J. Pattern Codification Strategies in Structured Light Systems. **Pattern Recognition**, v. 37, p. 827-849, 2004.

SANDU, D.; DEUGO, D. **The Lambda Pattern**. Proceedings of the 1999 Pattern Languages of Programming Conference. [S.l.]: [s.n.]. 1999.

SCHWARTZ, W. R. et al. Human detection using partial least squares analysis. **Computer Vision, 2009 IEEE 12th International Conference on**, p. 24-31, 2009. ISSN 978-1-4244-4420-5. Disponivel em: <http://www.umiacs.umd.edu/~lsd/papers/PLS-ICCV09.pdf>.

SCOTT, W. A. Reliability of Content Analysis: The Case of Nominal Scale Coding. **Public Opinion Quarterly**, v. 19, p. 321-325, 1955.

SENGHAS, A. A. K. S.; ÖZYÜREK, A. Children Creating Core Properties of Language: Evidence from an Emerging Sign Language in Nicaragua. **Science**, Washington, DC, 305, n. 5691, 2004. 1779-1782.

SHARP, T. **Implementing Decision Trees and Forests on a GPU**. Computer Vision – ECCV 2008. Berlin, Heidelberg: Springer Berlin / Heidelberg. 2008. p. 595-608.

SHOTTON, J. et al. **Real-Time Human Pose Recognition in Parts from Single Depth Images**. Computer Vision and Pattern Recognition. [S.l.]: [s.n.]. 2011.

SILVA, F. I. D. et al. **Curso de Pedagogia para Surdos: Língua Brasileira de Sinais**. Universidade do Estado de Santa Catarina. Florianópolis. 2002.

SOETENS, G. **Estimating the limitations of single-handed multi-touch input**. Utrecht University. Utrecht, The Netherlands. 2012.

SOUZA, C. R. Kernel Functions for Machine Learning Applications. **Personal Website**, 17 Março 2010. Disponivel em: <http://www.crsouza.com/2010/03/kernel-functions-for-machine-learning.html>. Acesso em: 01 nov. 2011.

SOUZA, C. R.; ANJO, M. S.; PIZZOLATO, E. B. **Fingerspelling Recognition with Support Vector Machines and Hidden Conditional Random Fields**. Proceedings of the 13th Ibero-American Conference on Artificial Intelligence (IBERAMIA 2012). Cartagena de Indias, Colombia: [s.n.]. 2012.

STARNER, T. **Visual Recognition of American Sign Language Using Hidden Markov Models**. Massachusetts Institute of Technology. [S.l.], p. 189-194. 1995.

STARNER, T.; WEAVER, J.; PENTLAND, A. Real-time American sign language recognition using desk and wearable computer based video. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, 20, n. 12, dec 1998. 1371 -1375.

STOKOE, W. C. **Language in hand:** why sign came before speech. [S.l.]: Gallaudet University Press, 2001.

STROBEL, K. **As imagens do outro sobre a cultura surda**. Florianópolis: Editora UFSC. 2008.

STROBEL, K. L.; FERNANDES, S. **Aspectos Linguísticos da LIBRAS**. Curitiba: SEED/SUED/DEE, 1998.

SUTTON, C.; MCCALLUM, A. Introduction to Statistical Relational Learning. In: GETOOR, L.; TASKAR, B. **An Introduction to Conditional Random Fields for Relational Learning**. [S.l.]: MIT Press, 2007.

SUTTON-SPENCE, R. A. W. B. **The linguistics of British Sign Language:** an introduction. [S.l.]: Cambridge University Press, 1999. ISBN 9780521637183.

TASKAR, B.; ABBEEL, P.; KOLLER, D. **Discriminative probabilistic models for relational data**. Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence. San Francisco, CA: Morgan Kaufmann Publishers. 2002. p. 485-492.

THE APACHE SOFTWARE FOUNDATION. Apache Commons Math, 2012. Disponivel em: <http://commons.apache.org/math>.

TSCHUPROW, A. A. **Principles of the mathematical theory of correlation, por A.A. Tschuprow; traduzido por M. Kantorowitsch**. [S.l.]: W. Hodge and Company, 1939. Disponivel em: <http://nla.gov.au/nla.cat-vn318152>.

VALENTINI, G.; DIETTERICH, T. G. Bias-Variance Analysis of Support Vector Machines for the Development of SVM-Based Ensemble Methods. **J. Mach. Learn. Res.**, v. 5, p. 725-775, December 2004. ISSN 1532-4435. Disponivel em: <http://dl.acm.org/citation.cfm?id=1005332.1016783>.

VALLI, C.; LUCAS, C. **Linguistics of American Sign Language:** an introduction. [S.l.]: Gallaudet University Press, 2000. ISBN 9781563680977.

VAPNIK, V. N. **The nature of statistical learning theory**. New York, NY, USA: Springer-Verlag New York, Inc., 1995. ISBN 0-387-94559-8.

VAPNIK, V. N. **Statistical learning theory**. [S.l.]: Wiley, 1998. ISBN 0471030031.

VIOLA, P.; JONES, M. **Robust Real-time Object Detection**. International Journal of Computer Vision. [S.l.]: [s.n.]. 2001.

VITERBI, A. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. **Information Theory, IEEE Transactions on**, v. 13, n. 2, p. 260 -269, april 1967. ISSN 0018-9448.

VOGLER, C.; METAXAS, D. A framework for recognizing the simultaneous aspects of American Sign Language. **Computer Vision and Image Understanding**, v. 81, p. 358-384, 2001.

WERBOS, P. J. **Beyond Regression:** New Tools for Prediction and Analysis in the Behavioral Sciences. [S.l.]: [s.n.], 1974. PhD thesis, Harvard University.

WHITEHILL, J.; OMLIN, C. W. **Haar Features for FACS AU Recognition**. Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition. Washington, DC, USA: IEEE Computer Society. 2006. p. 97-101.

WILLIAMS, L. **Spotting The Wisdom In the Crowds**. Imperial College London. London, UK. 2012.

WOLD, S.; SJÖSTRÖM, M.; ERIKSSON, L. PLS-regression: a basic tool of chemometrics. **Chemometrics and Intelligent Laboratory Systems**, v. 58, n. 2, p. 109-130, 2001. ISSN 0169-7439. Disponivel em: <http://www.sciencedirect.com/science/article/pii/S0169743901001551>.

WRIGHT, M. et al. **3D Gesture Recognition:** An Evaluation of User and System Performance. Pervasive Computing. [S.l.]: Springer Berlin / Heidelberg. 2011. p. 294-313.

WU, T.-F.; LIN, C.-J.; WENG, R. C. Probability Estimates for Multi-class Classification by Pairwise Coupling. **Journal of Machine Learning Research**, v. 5, p. 975-1005, 2003.

XUE, J.-H.; TITTERINGTON, D. M. Comment on "On Discriminative vs. Generative Classifiers: A Comparison of Logistic Regression and Naive Bayes". **Neural Process. Lett.**, v. 28, n. 3, p. 169-187, December 2008. ISSN 1370-4621. Disponivel em: <http://dl.acm.org/citation.cfm?id=1466642.1466648>.

YAMATO, J.; OHYA, J.; ISHII, K. Recognizing human action in time-sequential images using hidden Markov model. **Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition**, p. 379-385, 1992. Disponivel em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=223161>.

YANG, H.-D.; SCLAROFF, S.; LEE, S.-W. Sign Language Spotting with a Threshold Model Based on Conditional Random Fields. **IEEE Trans. Pattern Anal. Mach. Intell.**, Washington, DC, USA, v. 31, n. 7, p. 1264-1277, July 2009. ISSN 0162-8828. Disponivel em: <http://dl.acm.org/citation.cfm?id=1550409.1550658>.

YANG, R.; SARKAR, S. **Detecting Coarticulation in Sign Language using Conditional Random Fields**. Pattern Recognition, 2006. ICPR 2006. 18th International Conference on. [S.l.]: [s.n.]. 2006. p. 108-112.

ZAVALNIJS, A. **Improving Interaction in a Musical Tutor for Playing by Ear**. University of Edinburgh. [S.l.]. 2011.

Apêndices

Apêndice A

Consentimento Livre e Esclarecido

O texto a seguir reproduz o termo de consentimento livre e esclarecido, assinado por todos os participantes do experimento realizado durante esta pesquisa.

Termo de Consentimento Livre e Esclarecido

Você está sendo convidado para participar da pesquisa Reconhecimento de Gestos da Língua Brasileira de Sinais Através de Máquinas de Vetores de Suporte e Campos Condicionais Aleatórios Ocultos. Este experimento visa coletar sequências de imagens de pessoas articulando sinais da Língua Brasileira de Sinais (LIBRAS) utilizando-se uma câmera e um sensor de profundidade. Você foi selecionado como voluntário e sua participação não é obrigatória.

O objetivo deste experimento é montar um banco de dados para treinamento e teste de modelos de inteligência artificial e aprendizado de máquina para o reconhecimento automático de línguas de sinais. Sua participação nesta pesquisa consistirá em articular alguns sinais selecionados da língua de sinais em frente a uma câmera e a um sensor de profundidade. Ao assinar este termo, você afirma que está plenamente de acordo com as afirmações abaixo:

Entendo que os benefícios que receberei deste experimento se limitam ao aprendizado do material apresentado. Eu compreendo que sou livre para realizar perguntas a qualquer momento, solicitar que qualquer informação relacionada à minha pessoa não seja incluída no experimento, ou comunicar minha desistência de participação. Eu entendo que participo livremente com o único intuito de contribuir para a criação de modelos computacionais para a interpretação de Línguas de Sinais. Este experimento não envolverá riscos aos participantes.

Estou ciente de que toda informação coletada neste experimento será confidencial, e meu nome ou quaisquer outros meios de identificação não serão divulgados. Da mesma forma, me comprometo a manter sigilo das técnicas e documentos apresentados que fazem parte do experimento. A qualquer momento durante o experimento sei que será possível desistir de minha participação e retirar este consentimento. Esta recusa não trará nenhum prejuízo em sua relação ao pesquisador ou com a instituição. Sei que após a coleta de dados, uma etapa de pré-processamento eliminará quaisquer imagens identificáveis do usuário, eliminando quaisquer informações sobre o rosto do participante registrado nas imagens, restando-se apenas informações sobre sua localização e orientação espacial.

Esta pesquisa poderá ser acompanhada entrando-se em contato com um dos pesquisadores responsáveis, listados a seguir:

César Roberto de Souza (+55 19 9235-1622 / cesar.souza@dc.ufscar.br)
Prof. Dr. Ednaldo Brigante Pizzolato (ednaldo@dc.ufscar.br)

Programa de Pós-Graduação em Ciência da Computação – PPG-CC/DC/UFSCar
Rod. Washington Luís, Km 235; Caixa Postal 676; 13565-905 São Carlos-SP

Não haverá despesas decorrentes da participação nesta pesquisa. Você receberá uma cópia deste termo onde consta o telefone e o endereço do pesquisador principal, podendo tirar suas dúvidas sobre o projeto e sua participação, agora ou a qualquer momento. Este estudo é patrocinado pelo Conselho Nacional de Pesquisa e Desenvolvimento (CNPq).

Ao preencher e assinar este formulário, eu afirmo ter plena ciência e consentimento com os termos acima expostos.

Nome (em letra de forma): _____ **Curso/Ano:** _____

Assinatura: _____

Data: ____/_____/_____

Apêndice B

Resultados expandidos para os experimentos realizados

Este apêndice lista diferentes matrizes de confusão e demais resultados obtidos nos experimentos apresentados no Capítulo 8.

B.1 Soletração simplificada em imagens de intensidade

Nesta seção, apresentaremos resultados expandidos para o primeiro experimento realizado durante esta pesquisa. Na pagina a seguir, a Tabela 9 apresenta a matriz de confusão para a melhor SVM encontrada na tarefa de classificação de gestos estáticos no primeiro experimento realizado nesta pesquisa. A Tabela 10 apresenta a matriz de confusão para a melhor ANN encontrada neste mesmo contexto.

A Tabela 11 apresenta a porcentagem de vetores de suporte alocados para resolução de cada um dos subproblemas de decisão binários neste mesmo experimento. A Tabela 12 exibe o número de vetores de suporte limitados no fator de complexidade C da máquina de decisão, o que nos fornece uma estimativa da dificuldade de separação entre cada um dos gestos estáticos pertencentes ao alfabeto manual. Dos resultados, podemos observar como as maiores dificuldades residem sob a desambiguação entre os pares (F, T), (C, Ç), (N, M), (K, V) e similares.

Tabela 9. Matriz de confusão para a melhor SVM encontrada no experimento com soletração.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	Ç
A	99,30%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
B	0,00%	98,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,30%
C	0,30%	0,00%	90,30	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	9,00%
D	0,00%	0,00%	0,00%	97,30%	0,00%	0,00%	0,70%	0,00%	1,00%	0,30%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%
E	0,00%	0,00%	0,00%	0,00%	98,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,30%
F	0,00%	0,30%	0,00%	1,30%	0,00%	93,70%	0,00%	1,70%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	2,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%
G	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	95,30%	0,30%	0,00%	0,00%	0,30%	2,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
H	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,30%	92,00%	0,00%	0,00%	1,00%	0,70%	0,00%	0,00%	0,00%	0,30%	0,00%	1,70%	0,00%	0,00%	2,70%	0,00%	0,00%	0,00%	0,00%	1,00%	0,00%
I	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	99,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%
J	0,00%	0,00%	0,00%	0,30%	0,70%	0,00%	0,00%	0,00%	3,00%	85,00%	0,00%	0,00%	1,70%	0,00%	0,00%	1,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	3,00%	0,30%	4,70%	0,00%
K	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	2,30%	0,00%	0,00%	89,30%	1,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	2,30%	0,00%	0,00%	2,30%	0,70%	0,00%	0,00%	1,00%	0,00%
L	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	3,00%	0,00%	0,00%	0,00%	0,00%	97,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
M	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	96,00%	1,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,70%	0,00%
N	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	3,00%	96,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,70%
O	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	98,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%
P	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	98,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,30%	0,00%
Q	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,30%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	97,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%
R	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	96,30%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	2,00%	0,00%
S	0,70%	0,30%	0,00%	0,00%	2,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	96,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%
T	0,00%	0,00%	0,00%	0,00%	0,00%	4,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	95,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,00%	0,00%
U	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	2,00%	0,00%	0,00%	95,30%	0,00%	0,00%	0,00%	0,00%	1,30%	0,00%
V	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	3,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,00%	94,00%	0,70%	0,00%	0,00%	0,70%	0,00%
W	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	98,30%	0,00%	0,00%	0,70%	0,00%
X	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	95,70%	0,00%	3,00%	0,00%
Y	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	1,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	98,00%	0,00%	0,00%
Z	0,00%	1,00%	0,00%	0,30%	0,00%	0,30%	0,70%	1,30%	0,00%	1,70%	1,00%	0,00%	1,30%	0,30%	0,00%	2,70%	0,30%	1,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,00%	0,00%	87,00%	0,00%
Ç	0,00%	0,00%	5,00%	0,00%	1,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	1,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	92,00%

Tabela 10. Matriz de confusão para a melhor ANN encontrada no experimento com soletração.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	Ç
A	96,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	0,00%	2,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%
B	0,00%	97,30%	0,00%	0,00%	0,70%	0,30%	0,00%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,30%
C	0,00%	0,00%	94,30%	0,00%	0,00%	0,30%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	4,30%
D	0,30%	0,00%	0,30%	92,30%	0,00%	1,00%	0,70%	0,30%	2,00%	1,00%	0,70%	0,30%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%
E	0,00%	0,00%	0,00%	0,00%	98,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	1,00%	0,00%	0,00%	0,30%
F	0,00%	0,30%	0,30%	1,30%	0,00%	89,70%	0,70%	0,00%	0,30%	0,70%	0,00%	0,30%	0,00%	0,00%	0,00%	1,00%	0,00%	0,00%	0,00%	4,30%	0,00%	0,00%	0,30%	0,30%	0,00%	0,00%	0,30%
G	0,00%	0,00%	0,00%	0,70%	0,00%	0,30%	93,00%	0,30%	0,30%	0,00%	0,30%	1,30%	0,00%	0,00%	0,00%	0,00%	0,00%	3,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%
H	0,00%	0,30%	0,00%	0,70%	0,30%	1,00%	1,30%	88,00%	0,00%	0,00%	0,70%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	1,00%	0,00%	0,70%	3,70%	1,00%	0,00%	0,00%	0,00%	1,00%	0,00%
I	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	98,70%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%
J	0,70%	0,00%	0,00%	0,30%	1,00%	0,00%	0,00%	0,30%	1,30%	78,70%	0,00%	0,00%	3,00%	0,30%	0,30%	1,00%	1,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	2,70%	1,70%	7,00%	0,00%
K	0,30%	1,00%	0,00%	0,30%	0,00%	0,00%	1,00%	1,70%	0,30%	0,30%	83,30%	0,30%	1,00%	0,00%	0,00%	0,70%	0,00%	0,30%	1,70%	0,00%	1,00%	3,70%	1,30%	0,30%	0,00%	0,70%	0,70%
L	0,00%	0,00%	0,00%	1,00%	0,00%	0,00%	1,70%	0,00%	0,00%	0,00%	0,00%	97,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
M	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	92,70%	4,00%	0,00%	0,00%	1,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,70%	0,00%
N	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	4,00%	93,70%	0,00%	0,00%	1,30%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%
O	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	98,30%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%
P	0,00%	0,00%	1,30%	0,30%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	96,70%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,30%	0,00%
Q	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,70%	0,00%	0,00%	0,30%	1,30%	0,00%	0,00%	95,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	1,00%	0,00%
R	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,30%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	96,30%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	1,30%	0,30%
S	1,00%	0,70%	0,00%	0,00%	1,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,70%	1,00%	0,00%	0,00%	0,00%	94,70%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%
T	0,00%	0,00%	0,00%	0,30%	0,30%	7,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	91,00%	0,00%	0,00%	1,00%	0,00%	0,00%	0,00%	0,00%
U	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	2,70%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	3,70%	0,00%	0,00%	91,70%	0,30%	0,00%	0,00%	0,00%	0,70%	0,00%
V	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	3,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,70%	94,00%	1,00%	0,00%	0,00%	0,70%	0,00%
W	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	1,00%	0,00%	0,30%	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	0,30%	0,70%	1,00%	94,30%	0,00%	0,70%	0,00%	0,00%
X	0,00%	0,00%	0,00%	0,70%	0,30%	0,00%	0,00%	0,00%	0,00%	0,70%	0,00%	0,00%	2,00%	0,00%	0,00%	1,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	92,70%	0,00%	2,00%	0,00%
Y	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	98,30%	0,30%	0,00%
Z	0,00%	1,00%	0,30%	2,30%	0,00%	0,30%	1,00%	1,30%	0,00%	3,30%	2,00%	0,00%	1,30%	0,30%	0,30%	2,00%	0,70%	1,30%	0,00%	0,70%	0,30%	0,30%	0,70%	2,70%	1,30%	76,30%	0,00%
Ç	0,00%	0,00%	6,30%	0,00%	2,00%	0,30%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,30%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,30%	0,00%	0,00%	90,00%

Tabela 11. Número de vetores de suporte necessários por subproblema de decisão para melhor SVM.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	Ç
A	0,00%	27,80%	25,20%	31,50%	33,80%	30,70%	30,30%	31,80%	28,20%	43,30%	31,70%	22,00%	31,00%	26,80%	29,80%	32,80%	27,00%	25,30%	44,30%	32,30%	26,50%	30,30%	37,50%	37,80%	28,80%	40,70%	31,70%
B	27,80%	0,00%	25,20%	34,30%	37,20%	38,00%	37,70%	47,30%	34,20%	40,80%	38,20%	24,00%	30,80%	26,50%	26,50%	33,00%	26,30%	34,70%	28,20%	38,00%	33,00%	31,20%	41,80%	39,50%	25,30%	47,00%	30,80%
C	25,20%	25,20%	0,00%	38,20%	32,30%	38,70%	33,20%	34,80%	29,50%	37,50%	31,70%	28,70%	25,50%	24,20%	45,50%	39,20%	25,70%	28,70%	27,80%	38,80%	25,30%	28,70%	33,20%	37,80%	27,20%	41,50%	54,50%
D	31,50%	34,30%	38,20%	0,00%	36,80%	53,20%	56,80%	59,00%	43,70%	53,20%	50,50%	39,00%	31,80%	29,70%	35,70%	44,50%	31,70%	44,30%	32,80%	55,50%	38,30%	42,20%	46,20%	47,30%	38,70%	60,00%	41,50%
E	33,80%	37,20%	32,30%	36,80%	0,00%	41,00%	37,30%	41,30%	35,30%	43,80%	38,30%	27,20%	37,00%	30,80%	32,50%	34,50%	29,00%	32,30%	40,70%	40,30%	31,00%	32,00%	39,80%	43,70%	26,70%	45,80%	46,00%
F	30,70%	38,00%	38,70%	53,20%	41,00%	0,00%	49,50%	57,30%	43,50%	53,70%	44,00%	39,50%	34,00%	31,30%	37,00%	43,80%	33,30%	37,70%	30,80%	78,70%	34,80%	41,30%	49,30%	47,20%	40,70%	54,70%	47,30%
G	30,30%	37,70%	33,20%	56,80%	37,30%	49,50%	0,00%	66,00%	41,70%	51,70%	52,00%	42,80%	32,00%	29,30%	35,50%	41,20%	31,50%	50,00%	29,70%	50,00%	43,30%	44,70%	50,50%	46,00%	36,70%	59,50%	40,70%
H	31,80%	47,30%	34,80%	59,00%	41,30%	57,30%	66,00%	0,00%	41,20%	60,80%	62,30%	38,70%	35,30%	34,30%	37,70%	44,20%	34,80%	62,50%	33,00%	58,00%	62,50%	50,00%	59,70%	53,80%	41,50%	69,80%	43,50%
I	28,20%	34,20%	29,50%	43,70%	35,30%	43,50%	41,70%	41,20%	0,00%	47,30%	39,80%	27,20%	27,20%	27,20%	27,50%	34,70%	27,20%	34,30%	29,50%	39,50%	30,50%	33,70%	39,20%	39,70%	31,70%	46,70%	31,20%
J	43,30%	40,80%	37,50%	53,20%	43,80%	53,70%	51,70%	60,80%	47,30%	0,00%	56,50%	35,80%	51,80%	50,20%	41,80%	52,80%	49,30%	45,20%	38,80%	55,30%	42,30%	54,00%	60,30%	72,30%	52,70%	81,30%	44,50%
K	31,70%	38,20%	31,70%	50,50%	38,30%	44,00%	52,00%	62,30%	39,80%	56,50%	0,00%	34,20%	33,20%	34,00%	36,70%	39,80%	31,30%	49,00%	32,30%	48,20%	43,70%	65,50%	61,80%	50,20%	38,00%	59,80%	36,00%
L	22,00%	24,00%	28,70%	39,00%	27,20%	39,50%	42,80%	38,70%	27,20%	35,80%	34,20%	0,00%	23,80%	22,30%	27,20%	33,50%	23,70%	31,00%	23,50%	39,30%	25,70%	31,20%	34,30%	35,30%	27,80%	40,00%	30,70%
M	31,00%	30,80%	25,50%	31,80%	37,00%	34,00%	32,00%	35,30%	27,20%	51,80%	33,20%	23,80%	0,00%	61,70%	28,50%	37,20%	42,50%	28,50%	30,70%	33,30%	27,00%	31,80%	37,50%	52,50%	28,80%	51,00%	30,80%
N	26,80%	26,50%	24,20%	29,70%	30,80%	31,30%	29,30%	34,30%	27,20%	50,20%	34,00%	22,30%	61,70%	0,00%	27,70%	33,50%	46,50%	29,50%	26,80%	33,00%	27,00%	30,30%	35,70%	51,70%	29,50%	49,00%	28,20%
O	29,80%	26,50%	45,50%	35,70%	32,50%	37,00%	35,50%	37,70%	27,50%	41,80%	36,70%	27,20%	28,50%	27,70%	0,00%	40,30%	25,30%	29,80%	32,00%	38,20%	28,30%	32,00%	38,50%	41,80%	30,20%	43,50%	42,80%
P	32,80%	33,00%	39,20%	44,50%	34,50%	43,80%	41,20%	44,20%	34,70%	52,80%	39,80%	33,50%	37,20%	33,50%	40,30%	0,00%	32,30%	36,30%	31,80%	45,00%	33,80%	37,30%	45,20%	56,00%	34,30%	63,80%	40,30%
Q	27,00%	26,30%	25,70%	31,70%	29,00%	33,30%	31,50%	34,80%	27,20%	49,30%	31,30%	23,70%	42,50%	46,50%	25,30%	32,30%	0,00%	28,70%	25,20%	32,00%	28,00%	31,00%	33,30%	51,80%	29,50%	46,80%	27,00%
R	25,30%	34,70%	28,70%	44,30%	32,30%	37,70%	50,00%	62,50%	34,30%	45,20%	49,00%	31,00%	28,50%	29,50%	29,80%	36,30%	28,70%	0,00%	27,80%	43,50%	60,80%	39,50%	43,00%	45,00%	32,00%	58,80%	32,50%
S	44,30%	28,20%	27,80%	32,80%	40,70%	30,80%	29,70%	33,00%	29,50%	38,80%	32,30%	23,50%	30,70%	26,80%	32,00%	31,80%	25,20%	27,80%	0,00%	30,30%	27,00%	29,30%	36,80%	37,30%	26,20%	37,80%	32,70%
T	32,30%	38,00%	38,80%	55,50%	40,30%	78,70%	50,00%	58,00%	39,50%	55,30%	48,20%	39,30%	33,30%	33,00%	38,20%	45,00%	32,00%	43,50%	30,30%	0,00%	36,30%	43,20%	50,20%	49,50%	43,00%	61,30%	44,70%
U	26,50%	33,00%	25,30%	38,30%	31,00%	34,80%	43,30%	62,50%	30,50%	42,30%	43,70%	25,70%	27,00%	27,00%	28,30%	33,80%	28,00%	60,80%	27,00%	36,30%	0,00%	39,00%	43,70%	43,00%	30,70%	52,30%	28,80%
V	30,30%	31,20%	28,70%	42,20%	32,00%	41,30%	44,70%	50,00%	33,70%	54,00%	65,50%	31,20%	31,80%	30,30%	32,00%	37,30%	31,00%	39,50%	29,30%	43,20%	39,00%	0,00%	56,70%	46,80%	36,70%	56,30%	33,80%
W	37,50%	41,80%	33,20%	46,20%	39,80%	49,30%	50,50%	59,70%	39,20%	60,30%	61,80%	34,30%	37,50%	35,70%	38,50%	45,20%	33,30%	43,00%	36,80%	50,20%	43,70%	56,70%	0,00%	56,80%	44,30%	62,00%	40,70%
X	37,80%	39,50%	37,80%	47,30%	43,70%	47,20%	46,00%	53,80%	39,70%	72,30%	50,20%	35,30%	52,50%	51,70%	41,80%	56,00%	51,80%	45,00%	37,30%	49,50%	43,00%	46,80%	56,80%	0,00%	44,70%	79,30%	40,80%
Y	28,80%	25,30%	27,20%	38,70%	26,70%	40,70%	36,70%	41,50%	31,70%	52,70%	38,00%	27,80%	28,80%	29,50%	30,20%	34,30%	29,50%	32,00%	26,20%	43,00%	30,70%	36,70%	44,30%	44,70%	0,00%	49,30%	29,30%
Z	40,70%	47,00%	41,50%	60,00%	45,80%	54,70%	59,50%	69,80%	46,70%	81,30%	59,80%	40,00%	51,00%	49,00%	43,50%	63,80%	46,80%	58,80%	37,80%	61,30%	52,30%	56,30%	62,00%	79,30%	49,30%	0,00%	45,50%
Ç	31,70%	30,80%	54,50%	41,50%	46,00%	47,30%	40,70%	43,50%	31,20%	44,50%	36,00%	30,70%	30,80%	28,20%	42,80%	40,30%	27,00%	32,50%	32,70%	44,70%	28,80%	33,80%	40,70%	40,80%	29,30%	45,50%	0,00%

Tabela 12. Número de vetores de suporte limitados no fator de complexidade C para a melhor SVM.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	Ç
A	0,00%	0,20%	0,30%	0,30%	1,00%	0,20%	0,20%	0,70%	0,70%	0,80%	0,50%	0,00%	0,30%	0,00%	0,00%	0,00%	0,20%	0,20%	2,80%	0,20%	0,00%	0,00%	0,50%	0,20%	0,70%	0,70%	0,50%
B	0,20%	0,00%	0,50%	0,50%	1,30%	1,50%	1,00%	1,20%	0,70%	1,50%	0,50%	0,30%	0,30%	0,30%	0,00%	0,20%	0,30%	0,80%	0,00%	1,20%	0,70%	0,70%	0,80%	0,50%	0,20%	1,00%	0,80%
C	0,30%	0,50%	0,00%	0,50%	0,80%	2,20%	0,50%	0,70%	0,30%	0,70%	0,50%	0,30%	0,20%	0,00%	2,30%	1,00%	0,00%	0,30%	0,30%	1,20%	0,00%	0,20%	0,50%	0,30%	0,00%	0,50%	14,20%
D	0,30%	0,50%	0,50%	0,00%	0,00%	1,70%	3,30%	2,80%	1,50%	2,00%	1,80%	1,30%	0,20%	0,20%	0,00%	0,70%	0,00%	2,80%	0,00%	2,00%	1,30%	0,80%	0,70%	0,50%	0,70%	4,00%	0,30%
E	1,00%	1,30%	0,80%	0,00%	0,00%	0,50%	0,20%	0,00%	0,30%	0,70%	0,20%	0,00%	0,20%	0,50%	1,00%	0,20%	0,00%	0,00%	1,70%	0,20%	0,00%	0,00%	0,20%	0,50%	0,00%	1,20%	2,50%
F	0,20%	1,50%	2,20%	1,70%	0,50%	0,00%	1,30%	1,50%	1,20%	0,80%	0,50%	0,30%	0,20%	0,20%	0,30%	0,30%	0,20%	0,50%	0,20%	16,30%	0,50%	0,00%	0,70%	0,30%	0,70%	1,20%	1,80%
G	0,20%	1,00%	0,50%	3,30%	0,20%	1,30%	0,00%	2,70%	0,80%	1,70%	3,20%	2,30%	0,20%	0,00%	0,00%	0,50%	0,00%	2,70%	0,00%	1,20%	1,70%	1,30%	1,00%	0,30%	0,80%	4,20%	0,50%
H	0,70%	1,20%	0,70%	2,80%	0,00%	1,50%	2,70%	0,00%	1,00%	1,50%	4,30%	1,20%	0,70%	0,30%	0,30%	0,50%	0,20%	5,80%	0,00%	1,30%	8,70%	2,00%	1,70%	0,70%	1,00%	2,80%	0,70%
I	0,70%	0,70%	0,30%	1,50%	0,30%	1,20%	0,80%	1,00%	0,00%	3,00%	0,30%	0,00%	0,20%	0,00%	0,20%	0,00%	0,00%	0,00%	0,50%	0,80%	0,20%	0,30%	1,00%	0,30%	0,80%	1,30%	0,50%
J	0,80%	1,50%	0,70%	2,00%	0,70%	0,80%	1,70%	1,50%	3,00%	0,00%	1,00%	0,00%	4,80%	2,70%	1,00%	1,20%	2,80%	0,50%	0,70%	0,70%	1,20%	1,20%	1,80%	5,30%	2,00%	11,80%	1,20%
K	0,50%	0,50%	0,50%	1,80%	0,20%	0,50%	3,20%	4,30%	0,30%	1,00%	0,00%	0,70%	0,70%	0,00%	0,30%	0,50%	0,00%	1,80%	0,50%	0,80%	2,70%	9,80%	4,30%	0,30%	0,50%	3,20%	0,50%
L	0,00%	0,30%	0,30%	1,30%	0,00%	0,30%	2,30%	1,20%	0,00%	0,00%	0,70%	0,00%	0,00%	0,00%	0,00%	0,20%	0,00%	0,50%	0,00%	1,00%	0,00%	0,00%	0,30%	0,00%	0,30%	0,30%	0,20%
M	0,30%	0,30%	0,20%	0,20%	0,20%	0,20%	0,20%	0,70%	0,20%	4,80%	0,70%	0,00%	0,00%	9,50%	0,30%	0,80%	3,70%	0,20%	0,30%	0,30%	0,00%	0,20%	1,00%	3,50%	0,30%	4,00%	0,80%
N	0,00%	0,30%	0,00%	0,20%	0,50%	0,20%	0,00%	0,30%	0,00%	2,70%	0,00%	0,00%	9,50%	0,00%	0,00%	0,50%	4,80%	0,20%	0,30%	0,00%	0,20%	0,00%	0,20%	2,70%	0,00%	2,00%	0,30%
O	0,00%	0,00%	2,30%	0,00%	1,00%	0,30%	0,00%	0,30%	0,20%	1,00%	0,30%	0,00%	0,30%	0,00%	0,00%	0,20%	0,00%	0,00%	0,00%	0,20%	0,00%	0,20%	0,30%	0,50%	0,30%	0,30%	2,70%
P	0,00%	0,20%	1,00%	0,70%	0,20%	0,30%	0,50%	0,50%	0,00%	1,20%	0,50%	0,20%	0,80%	0,50%	0,20%	0,00%	0,50%	0,00%	0,20%	1,00%	0,20%	0,00%	0,30%	0,50%	0,50%	2,80%	0,80%
Q	0,20%	0,30%	0,00%	0,00%	0,00%	0,20%	0,00%	0,20%	0,00%	2,80%	0,00%	0,00%	3,70%	4,80%	0,00%	0,50%	0,00%	0,20%	0,20%	0,00%	0,20%	0,00%	0,20%	2,00%	0,20%	2,20%	0,30%
R	0,20%	0,80%	0,30%	2,80%	0,00%	0,50%	2,70%	5,80%	0,00%	0,50%	1,80%	0,50%	0,20%	0,20%	0,00%	0,00%	0,20%	0,00%	0,00%	0,80%	4,80%	0,50%	0,30%	0,20%	0,50%	3,50%	0,70%
S	2,80%	0,00%	0,30%	0,00%	1,70%	0,20%	0,00%	0,00%	0,50%	0,70%	0,50%	0,00%	0,30%	0,30%	0,00%	0,20%	0,20%	0,00%	0,00%	0,00%	0,20%	0,00%	0,20%	0,20%	0,00%	0,50%	1,00%
T	0,20%	1,20%	1,20%	2,00%	0,20%	16,30%	1,20%	1,30%	0,80%	0,70%	0,80%	1,00%	0,30%	0,00%	0,20%	1,00%	0,00%	0,80%	0,00%	0,00%	0,50%	0,20%	1,20%	0,30%	1,00%	1,20%	0,80%
U	0,00%	0,70%	0,00%	1,30%	0,00%	0,50%	1,70%	8,70%	0,20%	1,20%	2,70%	0,00%	0,00%	0,20%	0,00%	0,20%	0,20%	4,80%	0,20%	0,50%	0,00%	1,30%	1,30%	0,20%	0,00%	2,00%	0,00%
V	0,00%	0,70%	0,20%	0,80%	0,00%	0,00%	1,30%	2,00%	0,30%	1,20%	9,80%	0,00%	0,20%	0,00%	0,20%	0,00%	0,00%	0,50%	0,00%	0,20%	1,30%	0,00%	2,30%	0,20%	0,50%	1,80%	0,30%
W	0,50%	0,80%	0,50%	0,70%	0,20%	0,70%	1,00%	1,70%	1,00%	1,80%	4,30%	0,30%	1,00%	0,20%	0,30%	0,30%	0,20%	0,30%	0,20%	1,20%	1,30%	2,30%	0,00%	0,30%	0,70%	1,70%	0,50%
X	0,20%	0,50%	0,30%	0,50%	0,50%	0,30%	0,30%	0,70%	0,30%	5,30%	0,30%	0,00%	3,50%	2,70%	0,50%	0,50%	2,00%	0,20%	0,20%	0,30%	0,20%	0,20%	0,30%	0,00%	0,80%	5,70%	0,50%
Y	0,70%	0,20%	0,00%	0,70%	0,00%	0,70%	0,80%	1,00%	0,80%	2,00%	0,50%	0,30%	0,30%	0,00%	0,30%	0,50%	0,20%	0,50%	0,00%	1,00%	0,00%	0,50%	0,70%	0,80%	0,00%	1,50%	0,00%
Z	0,70%	1,00%	0,50%	4,00%	1,20%	1,20%	4,20%	2,80%	1,30%	11,80%	3,20%	0,30%	4,00%	2,00%	0,30%	2,80%	2,20%	3,50%	0,50%	1,20%	2,00%	1,80%	1,70%	5,70%	1,50%	0,00%	1,70%
Ç	0,50%	0,80%	14,20%	0,30%	2,50%	1,80%	0,50%	0,70%	0,50%	1,20%	0,50%	0,20%	0,80%	0,30%	2,70%	0,80%	0,30%	0,70%	1,00%	0,80%	0,00%	0,30%	0,50%	0,50%	0,00%	1,70%	0,00%

B.2 Palavras naturais em imagens de profundidade

Nas páginas seguintes, apresentaremos resultados expandidos para o segundo experimento realizado nesta pesquisa. A Tabela 15 exibe diferentes estatísticas de desempenho para as diversas combinações de modelos utilizadas nesta pesquisa.

Dentre as estatísticas apresentadas, reportamos o número máximo e mínimo de acertos na matriz de confusão para uma dada combinação de classificadores; a estatística κ de Cohen, o erro padrão e a variância para esta estatística; o erro padrão e a variância desta estatística sob a hipótese nula; o coeficiente τ de Kendall, o coeficiente φ de Pearson. Na segunda seção desta mesma tabela apresentamos a estatística χ^2, o coeficiente T de Tschuprow; o coeficiente C de Pearson, os coeficientes V de Sakoda e Cramer; e finalmente os índices de concordância geral (acurácia), concordância geométrica e concordância atribuída ao acaso.

Nas demais páginas, apresentamos as matrizes de confusão finais para o sistema de reconhecimento de gestos avaliado sob as 13 palavras naturais consideradas nesta pesquisa, disponíveis na Tabela 16, para máquinas quadráticas; na Tabela 17 para máquinas lineares; e Tabela 18 para modelos que consideraram apenas a trajetória dos movimentos para efetuar a classificação.

Tabela 13. Matrix de confusão para SVMs com função kernel quadrática para classificação das configurações de mãos no experimento com imagens de profundidade.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46
1	149	6	31	4	14	2	4	7	0	1	2	0	16	6	1	0	0	2	0	4	5	6	2	1	0	1	5	4	1	2	1	0	1	1	0	2	0	1	1	1	3	3	0	0	0	10
2	3	184	9	11	4	4	0	6	1	1	6	3	3	6	3	3	3	1	2	3	1	3	2	2	2	2	1	4	3	2	3	0	5	0	0	1	1	0	0	2	3	4	0	1	0	2
3	14	9	194	0	6	1	4	6	2	1	3	0	2	7	0	0	1	0	2	1	5	13	3	1	1	0	1	3	5	2	1	0	0	0	0	1	0	0	0	0	0	2	0	0	2	7
4	0	6	3	143	4	12	3	4	7	5	5	9	6	10	3	0	4	3	1	2	2	6	2	0	1	2	2	2	2	0	3	1	16	2	1	2	0	1	3	0	14	2	1	2	1	2
5	1	0	5	0	152	4	30	36	3	1	17	1	1	7	0	0	0	0	0	0	5	5	11	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	15
6	0	12	3	4	7	169	12	9	2	4	5	14	0	6	0	0	2	2	5	0	1	1	1	1	0	0	7	1	0	0	2	1	4	7	0	0	0	1	0	0	9	0	0	1	1	6
7	2	0	7	0	22	3	167	33	2	0	4	0	3	6	0	0	0	0	0	0	19	13	5	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	11
8	1	0	6	1	16	0	32	176	1	1	11	0	2	4	1	0	0	0	0	0	8	16	8	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	12
9	0	4	1	0	0	1	2	3	143	15	19	10	2	4	0	1	1	1	1	6	1	5	3	1	0	3	17	4	3	4	3	1	2	4	2	8	1	0	2	1	2	10	1	1	6	1
10	0	1	3	2	4	6	1	2	11	138	19	20	1	2	0	1	2	2	4	5	5	4	1	3	1	0	13	0	0	3	2	1	4	0	2	8	1	0	2	0	1	6	0	4	11	4
11	1	0	5	2	9	3	9	6	11	11	111	5	8	14	0	0	0	0	1	2	7	10	10	2	0	0	28	2	6	8	0	2	8	2	2	3	0	1	0	0	5	3	0	0	3	0
12	0	3	0	4	1	4	0	0	4	23	10	135	1	4	1	3	1	4	4	4	4	1	3	1	0	1	12	1	4	2	3	0	3	6	5	12	1	2	1	0	5	4	2	7	13	1
13	6	2	0	6	1	3	7	4	2	3	11	0	166	19	1	1	7	4	0	4	15	10	6	3	0	0	2	2	0	2	1	2	2	0	0	2	0	0	0	0	0	0	0	2	0	4
14	1	0	3	3	4	3	10	12	7	0	8	0	16	130	0	4	5	3	0	2	15	25	14	2	1	1	4	2	0	2	1	3	0	1	0	0	0	0	2	1	7	1	0	0	0	7
15	0	1	2	2	0	1	0	0	0	1	0	0	1	3	168	10	3	4	3	3	0	1	1	6	5	15	0	1	0	1	10	3	2	4	5	4	10	3	2	11	0	2	6	4	2	0
16	1	0	1	1	0	0	0	0	0	0	1	3	2	1	9	177	8	10	5	5	0	0	1	5	11	6	0	2	0	2	1	5	6	2	4	3	5	2	2	7	2	0	4	1	5	0
17	0	3	0	5	1	4	2	1	1	1	5	4	8	9	3	7	110	19	2	7	6	7	1	13	9	6	5	2	0	5	2	15	7	1	1	1	4	3	6	2	1	1	2	1	3	4
18	0	3	0	1	1	0	0	1	1	0	5	0	15	6	4	9	15	117	2	1	2	6	2	17	19	7	4	3	0	6	5	9	2	1	0	2	9	4	4	8	0	0	7	0	1	1
19	1	0	0	3	0	2	0	0	2	2	2	6	2	1	0	6	3	4	154	8	1	1	0	4	3	2	2	3	0	0	7	2	6	5	4	8	8	5	20	8	0	2	12	1	0	0
20	2	0	4	0	3	2	5	2	0	1	8	7	7	7	1	0	7	4	8	137	6	2	4	2	3	6	13	6	9	0	3	8	2	1	6	5	2	1	4	0	0	2	4	0	2	4
21	4	0	4	2	4	5	28	13	2	1	0	2	9	14	0	0	2	1	0	1	142	32	12	1	0	0	4	4	0	1	0	0	1	0	0	0	0	0	0	0	1	1	1	0	0	8
22	2	1	8	2	4	2	11	13	2	0	2	3	8	24	0	0	0	2	0	2	27	140	17	1	0	1	5	2	5	2	0	0	3	0	0	0	0	0	0	0	2	0	0	0	0	9
23	3	0	5	2	8	8	10	10	7	0	23	0	5	16	0	1	1	2	0	3	23	24	124	0	1	0	9	1	6	0	0	0	1	0	0	0	0	0	0	0	3	0	0	0	0	4
24	0	2	2	2	1	1	0	2	1	2	3	0	3	5	0	1	14	10	5	9	3	2	4	121	16	25	1	0	0	2	8	2	4	2	2	8	5	8	4	8	2	1	3	3	3	0
25	0	0	1	3	0	0	0	0	0	4	3	0	2	5	4	6	6	6	0	3	0	0	3	14	143	32	2	2	1	2	6	4	3	3	4	7	6	2	2	8	2	3	3	0	4	1
26	0	1	1	3	0	0	0	0	1	3	4	7	3	3	1	6	8	11	2	3	2	2	1	9	30	122	3	5	0	5	7	4	5	1	5	5	7	4	6	4	4	2	8	2	0	0
27	0	1	4	5	5	4	7	4	7	10	22	8	2	3	0	0	1	4	0	3	5	4	5	0	2	1	136	8	16	2	2	6	1	2	0	1	0	0	1	0	1	7	1	2	3	4
28	0	3	3	3	2	1	2	3	3	6	5	1	3	5	0	1	2	1	5	7	2	8	3	1	1	1	1	145	10	31	3	2	4	4	2	4	3	1	0	1	3	6	0	1	3	4
29	7	3	7	0	6	2	4	10	1	4	9	0	6	6	1	0	2	1	2	10	5	7	9	2	1	0	16	12	130	8	1	3	1	1	0	0	0	0	1	0	3	9	1	0	2	7
30	1	1	3	2	0	1	0	0	8	3	7	1	2	7	0	0	0	3	4	2	0	10	2	5	0	0	9	32	15	156	1	1	2	1	0	2	0	1	2	0	2	9	1	2	2	0
31	0	1	0	2	0	0	0	0	1	0	1	2	1	0	2	3	1	4	9	4	1	0	0	6	12	14	0	4	1	2	147	7	6	9	5	7	8	3	5	13	0	1	9	8	1	0
32	0	0	2	8	0	1	0	1	3	0	4	4	10	4	1	3	10	7	6	13	1	1	2	13	1	4	4	2	9	2	11	141	6	1	2	2	5	3	4	4	0	0	2	1	2	0
33	1	3	0	15	2	2	0	1	3	5	8	8	6	3	1	1	5	1	4	4	2	4	3	3	1	3	4	5	2	4	5	1	139	7	4	4	2	3	5	2	9	2	1	5	7	0
34	0	1	1	7	3	8	2	0	4	4	5	10	3	1	1	1	0	2	2	0	0	4	2	1	2	1	2	4	0	0	1	1	10	181	5	10	1	4	0	1	3	1	3	6	1	1
35	0	2	0	3	0	0	0	0	2	3	1	3	0	4	0	1	1	4	5	3	1	0	0	3	7	4	1	2	0	1	14	1	13	9	129	9	30	7	3	13	0	0	2	12	7	0
36	0	0	1	1	1	4	1	0	0	5	2	5	4	0	1	1	3	3	4	2	2	1	0	6	2	4	2	3	0	3	2	6	7	3	15	166	7	8	7	3	2	0	1	5	7	0
37	0	1	0	3	0	0	0	0	5	0	0	1	3	0	4	1	2	3	1	3	0	0	0	3	8	8	1	0	0	0	7	2	6	5	25	11	162	8	3	14	0	0	1	8	1	0
38	1	0	0	2	0	0	0	0	0	0	0	0	2	3	5	4	5	9	3	5	0	1	1	8	7	6	0	1	0	2	3	3	4	5	7	11	26	147	5	8	2	3	4	3	4	0
39	0	1	1	3	0	1	0	0	3	2	2	2	0	2	3	3	1	1	27	6	1	0	3	7	1	4	4	7	5	5	6	4	2	3	3	10	8	10	150	3	0	0	2	1	3	0
40	1	1	0	4	0	1	0	0	2	2	0	3	0	0	4	3	2	5	6	1	0	0	0	11	16	9	1	1	2	1	6	4	8	6	17	8	22	10	1	125	0	1	4	8	4	0
41	2	0	1	4	6	2	2	10	1	6	4	5	2	5	1	1	1	3	0	2	1	3	4	1	6	4	7	4	1	8	3	1	4	16	2	13	2	0	4	1	152	1	1	1	0	2
42	2	3	5	0	3	1	2	1	8	3	7	4	3	12	0	0	6	4	2	9	2	2	3	0	1	2	12	20	12	17	2	7	5	1	3	2	1	1	4	3	3	117	0	0	1	4
43	1	2	5	0	1	0	0	1	0	1	4	1	2	4	2	5	6	15	11	4	0	0	0	6	9	12	1	4	0	1	8	11	4	2	4	3	8	8	14	7	1	2	127	1	0	2
44	0	3	0	6	0	3	1	0	1	4	3	5	0	2	1	0	1	2	3	1	1	0	0	1	0	4	3	0	1	3	9	1	7	8	17	4	7	1	2	8	1	1	0	177	8	0
45	3	5	0	3	1	3	0	0	13	7	8	8	3	2	1	2	1	2	5	7	1	2	3	3	2	1	5	4	2	3	3	2	11	6	8	3	3	2	7	3	6	2	3	13	128	0
46	8	1	18	1	3	8	26	15	2	1	6	1	4	12	0	1	3	2	0	3	23	9	8	0	1	0	6	1	4	1	0	1	0	1	0	0	0	1	0	0	1	0	0	0	0	128

Tabela 14. Matrix de confusão para melhor ANNs para classificação das configurações de mãos no experimento com imagens de profundidade.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46
1	119	7	17	0	5	6	5	10	2	1	5	2	7	11	0	0	2	4	0	1	5	16	5	0	1	1	4	4	10	4	1	3	3	1	0	1	1	2	2	2	7	6	0	1	1	15
2	3	148	8	9	4	7	1	0	1	3	3	3	3	3	5	5	2	2	1	3	3	3	4	3	2	2	0	9	5	6	3	3	6	3	2	1	5	5	2	2	3	6	3	2	0	3
3	23	5	132	1	13	2	6	18	1	1	2	1	1	6	2	0	0	1	0	5	6	8	5	1	2	1	2	4	12	1	1	2	1	0	0	2	0	4	3	0	1	7	1	1	0	15
4	3	14	4	109	6	11	1	4	5	0	5	7	7	5	2	2	5	5	3	2	2	6	1	5	1	5	1	5	7	2	5	3	7	6	1	1	5	7	2	0	10	5	2	4	3	4
5	9	1	5	5	105	3	19	20	4	1	20	1	3	11	0	0	2	2	0	3	6	16	13	0	0	1	8	1	15	0	0	2	2	0	0	0	0	0	2	1	7	3	3	0	0	6
6	2	9	4	9	9	134	7	5	2	2	6	10	1	5	0	1	2	1	7	2	2	0	1	2	3	1	4	4	6	4	1	2	7	6	1	0	3	2	4	0	11	7	1	0	2	8
7	5	0	8	0	20	9	104	30	2	0	9	1	3	12	0	0	1	2	0	0	9	12	15	0	0	0	3	4	11	4	0	1	0	0	0	0	0	1	2	0	6	3	0	0	1	22
8	19	1	9	2	17	6	19	119	2	0	3	0	4	17	1	0	0	0	0	1	10	12	13	0	1	0	3	5	13	2	0	1	0	0	0	0	0	2	0	0	9	2	0	0	0	7
9	6	5	2	4	4	4	1	4	103	17	7	8	2	2	0	3	2	6	1	2	2	4	3	0	3	2	10	12	13	8	2	8	2	0	4	2	1	3	6	0	4	5	7	2	11	3
10	2	3	2	5	5	10	1	2	12	98	12	17	0	5	1	3	1	2	3	3	3	5	2	1	2	0	10	7	6	7	1	5	9	1	1	8	0	2	4	5	2	12	1	7	10	2
11	8	4	2	4	11	7	5	8	3	9	65	6	5	11	0	1	4	4	2	4	5	9	19	4	1	0	24	6	12	8	2	7	2	4	0	1	1	2	1	1	12	2	2	0	4	8
12	3	2	1	3	1	4	1	0	7	28	7	87	2	6	1	1	2	5	3	5	2	2	4	6	1	2	9	2	13	0	9	2	4	5	7	6	0	4	5	2	11	10	4	12	5	4
13	9	6	4	6	2	4	6	5	3	1	1	4	107	17	1	1	3	9	0	4	13	9	3	3	3	0	2	3	14	2	5	4	2	0	1	2	1	6	3	0	2	9	2	2	1	15
14	6	1	4	3	7	6	12	19	2	3	7	1	13	85	2	3	6	8	0	2	13	22	8	4	2	7	2	3	9	6	2	4	3	2	0	0	1	1	2	1	4	3	4	0	1	6
15	2	12	0	5	0	3	1	0	3	1	3	4	0	3	125	16	5	8	5	6	0	0	0	5	9	9	1	4	5	1	6	5	4	2	7	1	9	3	4	7	5	3	4	3	0	1
16	4	2	1	2	0	2	0	3	3	2	0	3	3	4	4	130	9	14	8	3	2	1	1	2	11	7	2	5	8	2	5	4	3	1	5	0	6	5	3	6	0	1	18	1	4	0
17	6	10	5	5	3	6	2	1	2	1	4	2	11	8	9	6	64	14	2	7	1	3	2	11	7	9	5	12	6	1	1	9	4	1	2	1	5	9	9	1	2	7	6	2	6	10
18	4	6	1	6	1	2	1	4	3	1	5	4	5	7	4	12	13	63	2	7	4	3	6	14	7	10	3	4	3	2	4	8	4	1	1	2	12	13	10	12	1	6	8	1	5	5
19	2	1	3	6	0	1	0	0	5	5	0	4	1	4	1	3	3	4	109	17	1	1	0	7	10	3	1	9	3	0	8	3	4	1	2	2	6	6	27	5	5	3	15	6	2	1
20	6	2	11	3	7	4	2	2	5	3	9	3	3	9	1	2	10	3	8	70	2	8	1	4	4	5	12	8	23	1	1	16	0	0	2	0	3	3	7	4	4	6	16	0	0	7
21	8	1	5	5	12	2	20	6	5	2	3	1	17	18	1	0	2	2	0	9	86	20	9	3	1	3	4	5	13	3	0	4	1	0	0	0	1	2	4	0	2	2	1	0	1	16
22	9	0	16	2	7	5	14	12	4	0	8	0	8	22	1	0	1	5	0	0	22	87	19	0	0	0	7	6	12	2	0	6	1	0	0	0	1	3	4	0	2	3	1	0	2	8
23	12	0	4	3	11	6	11	14	2	2	9	1	5	17	1	2	2	3	0	3	17	19	73	1	1	3	13	8	8	7	1	3	1	1	0	0	0	0	5	0	10	6	4	0	3	8
24	4	4	5	2	3	2	3	1	2	1	5	0	6	8	3	11	8	14	3	6	3	3	3	86	16	20	1	5	6	1	2	2	4	2	3	3	6	6	6	6	4	7	5	0	5	4
25	3	4	0	9	0	3	0	3	3	3	3	3	0	4	5	10	6	13	1	4	1	1	1	14	87	21	1	10	7	3	4	6	6	2	2	1	9	5	7	6	5	8	7	1	8	0
26	4	3	1	3	2	4	0	2	5	6	9	5	0	2	3	14	6	13	2	8	2	1	2	17	24	67	3	5	10	4	8	3	4	1	6	4	8	5	10	5	2	2	7	4	2	2
27	4	2	1	6	7	7	3	5	9	16	16	11	1	4	0	1	2	4	2	8	4	6	5	1	1	5	73	5	23	1	1	6	3	3	2	4	0	4	3	1	11	8	5	0	9	7
28	5	5	8	3	6	0	0	1	3	7	13	2	1	4	0	2	1	2	3	5	6	4	2	1	4	5	10	109	13	22	2	3	3	0	1	5	1	2	3	2	4	11	4	3	3	6
29	11	0	10	1	11	4	5	6	7	2	12	0	5	7	0	2	3	3	2	7	6	7	12	1	1	2	15	13	90	13	0	6	2	0	1	1	0	0	1	1	2	14	4	0	5	5
30	7	1	7	1	4	5	2	2	8	4	14	3	6	10	0	3	3	3	7	3	2	1	4	1	3	1	8	30	12	90	1	2	5	3	1	2	0	1	5	0	6	17	2	2	6	2
31	3	7	3	2	0	2	0	2	0	2	4	5	2	1	4	3	13	6	12	5	1	1	0	2	9	3	1	7	4	2	107	7	6	4	6	1	7	5	14	12	3	4	8	7	2	1
32	1	4	3	8	0	2	0	3	4	2	3	4	10	5	4	2	15	11	7	8	4	1	3	5	6	3	7	7	11	3	7	79	8	0	3	1	4	7	14	5	1	6	9	3	5	2
33	0	11	2	12	3	9	2	1	4	6	5	6	2	4	3	1	4	9	4	8	1	2	3	2	11	5	3	5	1	3	6	6	76	5	2	2	7	7	8	7	14	3	6	9	9	1
34	1	5	8	8	3	10	2	0	8	2	6	7	0	2	1	4	2	1	1	6	0	1	2	1	6	4	2	6	2	1	8	6	3	124	7	9	2	4	3	2	6	5	5	7	5	2
35	3	4	5	7	0	1	0	1	4	7	2	10	0	1	2	4	1	5	6	7	1	0	0	6	2	5	2	4	3	1	11	0	9	6	82	4	21	2	12	9	4	2	9	15	19	1
36	1	3	5	2	0	4	0	2	6	7	5	6	1	4	3	3	3	9	5	2	0	1	2	5	5	4	8	6	5	1	6	9	3	7	5	96	8	7	8	3	11	7	9	2	9	2
37	3	1	4	3	0	3	0	0	3	1	4	9	4	1	5	4	7	8	0	2	1	0	0	16	9	13	1	2	2	1	8	3	4	6	12	4	101	8	5	14	3	2	5	9	9	0
38	1	2	2	1	2	0	2	1	2	0	1	4	0	4	3	6	5	7	1	2	1	1	2	11	13	10	4	6	2	2	8	8	2	4	8	11	13	97	9	9	7	2	10	3	8	3
39	2	3	7	2	6	3	2	0	6	3	6	4	0	3	2	4	1	4	40	15	0	2	1	2	5	5	2	11	5	4	6	3	6	0	1	3	6	5	92	4	1	8	7	4	2	2
40	0	4	3	3	1	2	0	0	1	2	1	6	0	3	7	5	2	12	5	3	1	0	1	6	11	7	2	2	5	1	10	10	6	4	15	1	18	11	13	84	4	2	12	6	8	0
41	3	5	3	5	7	6	2	7	2	3	7	10	1	7	0	2	5	2	1	4	0	3	3	3	4	4	7	1	8	4	1	5	3	2	1	3	0	3	5	0	142	3	7	0	3	3
42	3	3	8	4	2	5	2	6	11	3	8	5	5	10	0	1	3	3	2	9	2	3	4	5	3	3	9	20	14	5	2	6	5	1	1	1	1	3	8	4	6	90	3	1	3	4
43	0	4	7	5	2	4	0	0	0	3	4	3	1	5	3	7	13	7	10	8	2	1	2	5	8	6	3	6	6	1	6	4	4	2	3	1	3	6	13	7	5	4	106	2	4	4
44	1	7	1	5	1	5	1	1	3	4	3	10	0	1	2	2	2	4	4	1	0	2	2	6	4	1	2	1	1	8	6	8	7	9	4	6	9	4	6	1	4	4	130	14	1	
45	5	4	3	4	0	5	2	4	6	7	4	6	3	4	1	3	3	2	8	8	0	1	5	1	1	2	3	8	7	2	5	10	5	2	6	3	5	2	13	8	7	6	5	10	99	2
46	7	2	16	3	1	7	19	9	1	0	4	0	8	9	0	1	5	5	0	2	23	13	11	2	1	0	11	3	16	4	0	5	4	3	0	0	1	0	5	0	1	2	0	0	0	96

Tabela 15. Estatísticas de desempenho para diferentes combinações de modelos de classificação no experimento com palavras naturais em imagens de profundidade.

Rotulação	Classificação	Máximo de Acertos	Mínimo de Acertos	Cohen κ	SE(κ)	Var(κ)	SE(κ) H_0	Var(κ) H_0	Kendall τ	Pearson φ
SVM Quadrática	HMM	86	41	0,807	1,35E-02	1,83E-04	9,47E-03	8,97E-05	0,81	2,823
SVM Quadrática	HCRF	84	45	0,844	1,24E-02	1,54E-04	9,47E-03	8,97E-05	0,846	2,934
SVM Linear	HMM	84	36	0,797	1,38E-02	1,90E-04	9,47E-03	8,97E-05	0,8	2,806
SVM Linear	HCRF	85	39	0,82	1,32E-02	1,73E-04	9,49E-03	9,00E-05	0,822	2,865
ANN	HMM	85	35	0,749	1,49E-02	2,22E-04	9,48E-03	8,99E-05	0,754	2,662
ANN	HCRF	83	36	0,775	1,44E-02	2,06E-04	9,50E-03	9,02E-05	0,779	2,728
Nenhum	HMM	86	19	0,533	1,74E-02	3,04E-04	9,47E-03	8,97E-05	0,55	2,074
Nenhum	HCRF	85	31	0,554	1,74E-02	3,03E-04	9,48E-03	8,99E-05	0,569	2,114

Rotulação	Classificação	χ^2	Tschuprow T	Pearson C	Sakoda V	Cramer V	Concordância	Concordância Geométrica	Concordância ao acaso
SVM Quadrática	HMM	7485,838	0,016	0,943	0,981	0,815	0,822	58,392	0,078
SVM Quadrática	HCRF	8082,566	0,016	0,947	0,985	0,847	0,856	60,986	0,078
SVM Linear	HMM	7395,4	0,016	0,942	0,98	0,81	0,813	57,574	0,078
SVM Linear	HCRF	7708,32	0,016	0,944	0,983	0,827	0,834	59,211	0,078
ANN	HMM	6653,34	0,015	0,936	0,974	0,768	0,769	54,188	0,078
ANN	HCRF	6988,485	0,016	0,939	0,977	0,788	0,792	55,994	0,078
Nenhum	HMM	4037,461	0,014	0,901	0,938	0,599	0,57	38,363	0,078
Nenhum	HCRF	4197,171	0,014	0,904	0,941	0,61	0,589	40,829	0,078

Tabela 16. Matriz de confusão para máquinas quadráticas combinadas com HMMs e HCRFs para reconhecimento de palavras naturais da Libras em imagens de profundidade.

HMM - SVM Quadrática

	Armário	Carro	Comprar	Desculpa	Dia	Eu	Gostar	Idade	Nome	Oi	Querer	Sapato	Tchau	Total
Armário	71	2	0	0	0	0	0	0	0	1	1	0	0	75
Carro	4	64	0	0	0	0	0	0	0	0	0	0	0	68
Comprar	1	0	64	0	0	0	6	0	5	3	5	4	0	88
Desculpa	0	0	0	86	0	0	0	0	0	0	0	0	0	86
Dia	0	0	0	0	56	0	1	1	0	0	1	0	2	61
Eu	0	0	1	0	1	51	6	3	0	1	5	1	0	69
Gostar	4	2	0	0	0	8	41	0	2	4	3	0	1	65
Idade	0	0	0	1	2	4	1	51	0	0	2	10	1	72
Nome	1	1	5	0	0	2	0	1	57	4	0	0	0	71
Oi	3	1	5	0	1	3	0	0	3	63	1	0	0	80
Querer	1	3	1	0	0	4	1	0	0	0	54	0	0	64
Sapato	2	1	2	0	0	0	0	7	1	0	2	66	0	81
Tchau	0	0	0	0	4	0	6	0	1	0	0	0	48	59
Total	87	74	78	87	64	72	62	63	69	76	74	81	52	939

HCRF - SVM Quadrática

	Armário	Carro	Comprar	Desculpa	Dia	Eu	Gostar	Idade	Nome	Oi	Querer	Sapato	Tchau	Total
Armário	70	2	0	0	0	0	1	0	0	1	1	0	0	75
Carro	2	62	0	0	0	3	0	0	0	0	1	0	0	68
Comprar	1	0	73	0	0	1	0	1	6	3	0	2	1	88
Desculpa	1	0	0	84	0	0	0	1	0	0	0	0	0	86
Dia	0	0	0	0	56	0	0	1	0	0	1	0	3	61
Eu	2	0	1	0	1	55	4	1	0	1	3	1	0	69
Gostar	2	0	0	0	0	7	45	1	1	2	2	1	4	65
Idade	0	0	0	0	1	3	2	60	0	0	1	3	2	72
Nome	0	0	1	0	0	0	1	1	63	5	0	0	0	71
Oi	2	0	4	0	1	4	2	0	1	64	0	1	1	80
Querer	1	2	1	0	0	5	2	0	0	0	49	1	3	64
Sapato	1	0	1	0	0	0	2	6	0	0	0	71	0	81
Tchau	0	0	0	0	5	0	1	0	1	0	0	0	52	59
Total	82	66	81	84	64	78	60	72	72	76	58	80	66	939

Tabela 17. Matriz de confusão para máquinas lineares combinadas com HMMs e HCRFs para reconhecimento de palavras naturais da Libras em imagens de profundidade.

HMM - SVM Linear

	Armário	Carro	Comprar	Desculpa	Dia	Eu	Gostar	Idade	Nome	Oi	Querer	Sapato	Tchau	Total
Armário	72	1	0	0	0	0	0	0	0	0	1	1	0	75
Carro	10	57	0	0	0	0	0	0	0	0	1	0	0	68
Comprar	1	0	69	0	0	1	3	0	6	3	4	1	0	88
Desculpa	1	0	0	84	0	0	0	0	0	0	0	1	0	86
Dia	0	0	0	0	58	0	0	0	0	2	0	1	0	61
Eu	0	0	0	0	1	49	9	2	0	1	6	1	0	69
Gostar	1	2	2	0	0	11	36	0	1	0	8	0	4	65
Idade	0	0	1	2	0	6	2	51	0	0	0	9	1	72
Nome	3	0	4	0	0	1	1	0	56	5	1	0	0	71
Oi	1	0	3	0	0	1	2	0	7	63	2	1	0	80
Querer	2	3	0	0	0	2	0	0	0	0	57	0	0	64
Sapato	2	0	3	0	0	1	2	11	0	1	1	60	0	81
Tchau	0	1	0	0	1	3	3	0	0	0	0	0	51	59
Total	93	64	82	86	60	75	58	64	70	75	81	75	56	939

HCRF - SVM Linear

	Armário	Carro	Comprar	Desculpa	Dia	Eu	Gostar	Idade	Nome	Oi	Querer	Sapato	Tchau	Total
Armário	64	7	0	0	0	0	0	0	0	0	2	2	0	75
Carro	9	57	0	0	0	0	0	0	0	1	1	0	0	68
Comprar	2	1	75	0	0	1	1	0	4	2	1	1	0	88
Desculpa	1	0	0	84	0	0	0	0	0	1	0	0	0	86
Dia	0	0	0	0	56	2	1	0	0	1	1	0	0	61
Eu	0	0	1	0	0	53	9	0	0	1	2	3	0	69
Gostar	3	3	3	0	1	6	39	1	1	2	2	0	4	65
Idade	0	0	0	0	0	5	3	50	2	2	1	9	0	72
Nome	1	0	5	0	0	0	1	0	59	4	0	1	0	71
Oi	4	0	2	0	0	2	2	0	2	66	1	1	0	80
Querer	1	4	2	0	1	1	1	0	0	2	51	1	0	64
Sapato	3	0	2	0	0	0	2	8	0	2	0	64	0	81
Tchau	0	0	0	0	3	1	2	0	1	1	0	0	51	59
Total	88	72	90	84	61	71	61	59	69	85	62	82	55	939

Tabela 18. Matriz de confusão para HMMs e HCRFs sem incorporar informações linguísticas para reconhecimento de palavras naturais da Libras em imagens de profundidade.

HMM - Somente Trajetória

	Armário	Carro	Comprar	Desculpa	Dia	Eu	Gostar	Idade	Nome	Oi	Querer	Sapato	Tchau	Total
Armário	54	17	0	0	0	0	1	0	1	1	1	0	0	75
Carro	20	41	0	1	0	0	1	2	1	0	0	2	0	68
Comprar	1	4	52	0	0	0	5	1	5	4	12	4	0	88
Desculpa	0	0	0	86	0	0	0	0	0	0	0	0	0	86
Dia	2	0	0	0	46	4	0	4	0	0	0	1	4	61
Eu	1	0	6	0	1	19	13	5	0	1	17	4	2	69
Gostar	2	1	1	0	0	12	26	2	2	13	2	3	1	65
Idade	0	0	7	2	4	3	0	35	1	1	4	13	2	72
Nome	1	2	6	0	0	1	1	0	39	18	0	1	2	71
Oi	3	1	9	1	1	0	2	0	34	25	3	0	1	80
Querer	0	3	1	0	1	5	4	2	1	7	39	1	0	64
Sapato	7	3	1	1	0	1	0	22	1	0	3	42	0	81
Tchau	0	0	0	3	18	1	4	1	1	0	0	0	31	59
Total	91	72	83	94	71	46	57	74	86	70	81	71	43	939

HCRF - Somente Trajetória

	Armário	Carro	Comprar	Desculpa	Dia	Eu	Gostar	Idade	Nome	Oi	Querer	Sapato	Tchau	Total
Armário	53	16	1	0	0	0	2	0	1	0	1	1	0	75
Carro	14	38	2	0	0	1	3	0	3	2	1	4	0	68
Comprar	0	0	53	0	0	2	2	1	8	11	8	3	0	88
Desculpa	0	0	0	85	0	0	0	0	1	0	0	0	0	86
Dia	2	0	0	0	41	2	2	4	0	0	0	0	10	61
Eu	1	0	5	0	2	33	8	8	0	1	7	4	0	69
Gostar	2	2	0	0	0	12	31	2	2	2	6	5	1	65
Idade	0	1	3	0	2	3	1	36	0	2	5	16	3	72
Nome	0	2	4	0	0	1	0	1	37	25	0	0	1	71
Oi	3	2	3	0	0	3	2	0	28	31	4	2	2	80
Querer	0	2	4	0	0	6	0	4	2	7	34	4	1	64
Sapato	6	8	4	0	0	5	0	15	0	0	3	40	0	81
Tchau	0	0	0	0	10	1	4	0	1	1	0	1	41	59
Total	81	71	79	85	55	69	55	71	83	82	69	80	59	939

Apêndice C

Um framework para suporte da aplicação

"Construímos muros demais e pontes de menos." — Sir Isaac Newton

FRAMEWORKS DE SOFTWARE são úteis não apenas para reusar o código principal de uma aplicação, mas para descrever os desenhos abstratos dos componentes de uma biblioteca (JOHNSON e FOOTE, 1988) e como eles se relacionam entre si. Nada mais justo para se organizar o conhecimento adquirido ao decorrer desta dissertação do que implantar o modelo mental das técnicas, dos componentes e do framework teórico associado ao reconhecimento de padrões e visão computacional em um framework de software descrito ao decorrer desta seção.

O *framework* de software aqui apresentado chama-se Accord.NET. Este framework é organizado como um framework de componentes, ou *framelets* (PREE e KOSKIMIES, 1999), de maneira que cada método possua sua própria coleção de pontos quentes (*hot spots*) e frios (*cold spots*). Dentro de cada *framelet*, um método que pode ser executado de diversas formas possui cada uma de suas implementações disponíveis através do emprego do padrão de projeto *estratégia;* como, por exemplo, as diferentes formas de se treinar um HCRF. Note que esta abordagem não é nova, sendo que a aplicação dos padrões *estratégia* e *método template* também são amplamente utilizados no *Apache Commons Math* (THE APACHE SOFTWARE FOUNDATION, 2012).

De fato, os princípios seguidos para o desenvolvimento deste framework foram os mesmos adotados pela *Apache Commons*: ênfase em componentes pequenos e facilmente integráveis, evitando dependências e configurações complexas; a adoção do padrão estratégia para múltiplas implementações de uma mesma técnica ou método; e a adoção apenas de um pequeno número limitado dependências. O emprego do padrão estratégia no framework proposto também foi bastante inspirado pela abordagem adotada pelo framework de software AForge.NET (KIRILLOV, 2012).

No entanto, o que distingue este framework dos demais é a concentração destes diversos métodos de aprendizado de máquina e visão computacional sob uma única embalagem consistente e de fácil acesso. A disponibilidade, por exemplo, de framelets básicos para especificação de distribuições de probabilidade e funções *kernel* possibilita que novos métodos e técnicas sejam implementados com facilidade. O emprego de técnicas como o padrão de projeto lambda (SANDU e DEUGO, 1999) possibilita tornar métodos disponíveis aplicáveis a novas estruturas de dados ou novas implementações sem que seja necessário assumir que estas implementações cumpram alguma determinada interface.

O *framework* foi construído tendo como base o framework AForge.NET, o que possibilita combinar de maneira fácil técnicas de processamento de imagens disponíveis deste framework com os métodos de aprendizado de máquina disponíveis no *framework* proposto; mais além, ainda foi possível compartilhar código entre os dois projetos de maneira que implementações originalmente pertencentes ao framework proposto foram integradas no framework original de Kirillov.

C.1 Casos de uso

Pode-se notar que nenhum esforço empregado na elaboração deste *framework* foi em vão, tendo em vista as diversas aplicações já produzidas a partir de seu uso. Como exemplo das diversas possibilidades de utilização deste framework, podemos destacar os trabalhos de Hassani (2011), que o utilizou para investigar a aceitação de idosos ao uso do sensor *Kinect* na interação entre humanos e robôs, o trabalho de Alosefer e Rana (2011) que o utilizou para construir algoritmos de predição de ataques via sites maliciosos através de analise comportamental de dados de *Honeypot*[16], a investigação de Wright *et al.* (2011) acerca de interfaces gestuais 3D e Zavalnijs (2011) que o utilizou no contexto de elaboração de um tutor virtual para estudantes de música habituados a aprender por reprodução auditiva.

Trabalhos mais recentes incluem a aplicação desenvolvida por Brown *et al.* (2012) que permite a um especialista encontrar uma função de distância apropriada para determinado problema através de uma inspeção visual interativa. Em seu trabalho, os autores utilizaram o framework para realizar a transformação de PCA do conjunto de dados sendo analisado. Outra aplicação interessante é a de Soetens

[16] *Honeypot* é uma terminologia utilizada para descrever armadilhas destinadas a detectar, defender ou contra-atacar tentativas de invasão em sistemas de informação.

(2012), que utilizou as capacidades de processamento matricial e de regressão logística para realizar sua investigação acerca das restrições espaciais na realização de gestos envolvendo múltiplos dedos em superfícies sensíveis a toque. Sua pesquisa possibilita, por exemplo, que a interação de uma aplicação se adeque as dimensões da mão do usuário.

O trabalho de Bursztein *et al.* (2011) realizou a avaliação da suscetibilidade de sistemas CAPTCHA baseados em caracteres distorcidos a ataques automatizados. Seu sistema de quebra de CAPTCHAS é baseado nos frameworks AForge.NET e Accord.NET, fazendo uso dos filtros de imagens e a implementação das técnicas de aprendizado de máquina disponível nestes *frameworks*. Já o trabalho de Lidegaard (2012) utilizou diversos *frameworks*, entre eles os dois anteriormente citados, na construção de um dispositivo para rastreio da direção do olhar em um ambiente 3D montável sobre a cabeça do usuário.

O uso do *framework* também tem se mostrado particularmente atraente para trabalhos finais de graduação, em que estudantes talvez disponham de um curto prazo de tempo para entrar em contato e estudar algum método específico para a aplicação em seu problema de investigação. Alguns destes projetos incluem o sistema de interação 3D de Lourenço (2010) utilizando o controle *WiiMote* para a plataforma de jogos *Nintendo Wii*, o trabalho de Mendelssohn (2010) envolvendo o reconhecimento de gestos em um quadro-branco digital e o trabalho de Brummitt (2011) para arbitrar jogos de *Scrabble*. Em um curso de mestrado de curta duração, Williams (2012) utilizou as técnicas de LDA, regressão logística e o classificador de Naïve Bayes disponíveis no *framework* para encontrar a sabedoria residente em multidões, filtrando o conselho de um extenso número de pessoas para se encontrar informações pertinentes.

Finalmente, podemos destacar alguns produtos construídos sobre o *framework*. O software *Harperia* desenvolvido por *Vladimir Plokhov* emprega os *frameworks* AForge.NET e Accord.NET para reconhecimento de áudio e voz em tempo real. O software *Point-and-Call* desenvolvido por *Antti Savolainen*, disponível para Windows Phone, é capaz de realizar chamadas apenas apontando a câmera do dispositivo para um número de telefone em uma superfície qualquer. A aplicação utiliza SVMs criadas utilizando-se o framework para detectar e reconhecer os dígitos na imagem capturada pela câmera, oferendo o número encontrado para ser discado logo em seguida.

C.2 Código livre

O *framework* apresentado encontra-se disponível sob a licença de software livre LGPL, o que possibilita seu uso com mínimas restrições. Sendo software livre, seu código está disponível na página do projeto[17], bem como o código de todas as aplicações que demonstram seu uso nas mais variadas tarefas em computação científica.

[17] http://accord.googlecode.com