UNIVERSIDADE DE SÃO PAULO

Escola de Artes, Ciências e Humanidades

Renata Cristina Barros Madeo

Máquinas de Vetores Suporte e a Análise de Gestos: incorporando aspectos temporais

São Paulo
Maio 2013

Renata Cristina Barros Madeo

Máquinas de Vetores Suporte e a Análise de Gestos: incorporando aspectos temporais

Dissertação apresentada à Escola de Artes, Ciências e Humanidades da Universidade de São Paulo para obtenção do título de Mestre em Ciências.

Versão corrigida contendo as alterações e as correções sugeridas pela banca examinadora. A versão original encontra-se disponível na Escola de Artes, Ciências e Humanidades.

Orientador: Profa. Dra. Sarajane Marques Peres

São Paulo
Maio 2013

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada à fonte.

CATALOGAÇÃO-NA-PUBLICAÇÃO
Biblioteca
Escola de Artes, Ciências e Humanidades da Universidade de São Paulo

Madeo, Renata Cristina Barros
Máquinas de vetores suporte e a análise de gestos : incorporando aspectos temporais / Renata Cristina Barros Madeo ; orientador[a] Sarajane Marques Peres. – São Paulo, 2013.
112 f. : il.

Dissertação (Mestrado em Ciências) – Programa de Pós-Graduação em Sistemas de Informação, Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, em 2013.
Versão corrigida.

1. Aprendizado computacional. 2. Inteligência artificial. 3. Análise do movimento humano. 4. Gestos. 5. Modelos matemáticos. I. Peres, Sarajane Marques, orient. II. Título.

CDD 22.ed. – 006.31

Dissertação de mestrado sob o título *"Máquinas de Vetores Suporte e a Análise de Gestos: incorporando aspectos temporais"*, defendida por Renata Cristina Barros Madeo e aprovada em 15 de maio de 2013, em São Paulo, Estado de São Paulo, pela banca examinadora constituída pelos doutores:

Profa. Dra. Sarajane Marques Peres
Orientadora

Prof. Dr. Fernando José Von Zuben
Universidade Estadual de Campinas

Prof. Dr. João Luiz Bernardes Júnior
Universidade de São Paulo

Dedico este trabalho aos meus pais, Delsa e João, pelo apoio e pelo amor que sempre me deram.

Em primeiro lugar, agradeço aos meus pais por todo o amor, incentivo e cuidado que me possibilitaram chegar até este momento.
Agradeço especialmente à minha orientadora, Profa. Dra. Sarajane Marques Peres, por toda a dedicação, apoio e orientação desde a graduação.
Também dedico um agradecimento especial ao Prof. Dr. Clodoaldo Lima pela atenção e por toda a ajuda com Máquinas de Vetores Suporte, processamento de sinais, Fourier e Wavelet.
Ao Prof. Dr. Leland McCleary, agradeço por nos apresentar a área de estudos dos gestos e pela sua ajuda ao longo do trabalho.
Aos professores do Programa de Pós-graduação em Sistemas de Informação que contribuíram para a minha formação.
Aos amigos do PPgSI e do GrIA, agradeço pela oportunidade de compartilhar bons momentos durante as aulas e as tardes no laboratório.
Aos alunos do PET-SI, especialmente à Priscilla, pela enorme ajuda na fase de aquisição de dados.
Ao meu namorado, Paulo, agradeço pelo carinho, paciência e compreensão durante todo o período do mestrado.
Agradeço também à FAPESP pelo financiamento desta pesquisa por meio do processo 2011/04608-8, à CAPES e à Universidade de São Paulo.
Por fim, agradeço a Deus pela minha vida e pela oportunidade de viver esta experiência ao lado de tantas pessoas especiais.

Resumo

Recentemente, tem se percebido um interesse maior da área de computação pela pesquisa em análise de gestos. Parte dessas pesquisas visa dar suporte aos pesquisadores da área de "estudos dos gestos", que estuda o uso de partes do corpo para fins comunicativos. Pesquisadores dessa área analisam gestos a partir de transcrições de conversas ou discursos gravados em vídeo. Para a transcrição dos gestos, geralmente realiza-se a sua segmentação em unidades gestuais e fases. O presente trabalho tem por objetivo desenvolver estratégias para segmentação automatizada das unidades gestuais e das fases dos gestos contidos em um vídeo no contexto de contação de histórias, formulando o problema como uma tarefa de classificação supervisionada. As Máquinas de Vetores Suporte foram escolhidas como método de classificação, devido à sua capacidade de generalização e aos bons resultados obtidos para diversos problemas complexos. Máquinas de Vetores Suporte, porém, não consideram os aspectos temporais dos dados, características que são importantes na análise dos gestos. Por esse motivo, este trabalho investiga métodos de representação temporal e variações das Máquinas de Vetores Suporte que consideram raciocínio temporal. Vários experimentos foram executados neste contexto para segmentação de unidades gestuais. Os melhores resultados foram obtidos com Máquinas de Vetores Suporte tradicionais aplicadas a dados janelados. Além disso, três estratégias de classificação multiclasse foram aplicadas ao problema de segmentação das fases dos gestos. Os resultados indicam que um bom desempenho para a segmentação de gestos pode ser obtido ao realizar o treinamento da estratégia com um trecho inicial do vídeo para obter uma segmentação automatizada do restante do vídeo. Assim, os pesquisadores da área de "estudos dos gestos" poderiam segmentar manualmente apenas um trecho do vídeo, reduzindo o tempo necessário para realizar a análise dos gestos presentes em gravações longas.

Palavras-chave: Análise de gestos, segmentação de unidades gestuais, segmentação das fases do gesto, Máquinas de Vetores Suporte.

Abstract

Recently, it has been noted an increasing interest from computer science for research on gesture analysis. Some of these researches aims at supporting researchers from "gesture studies", which studies the use of several body parts for communicative purposes. Researchers of "gesture studies" analyze gestures from transcriptions of conversations and discourses recorded in video. For gesture transcriptions, gesture unit segmentation and gesture phase segmentation are usually employed. This study aims to develop strategies for automated segmentation of gestural units and phases of gestures contained in a video in the context of storytelling, formulating the problem as a supervised classification task. Support Vector Machines were selected as classification method, because of its ability to generalize and good results obtained for many complex problems. Support Vector Machines, however, do not consider the temporal aspects of data, characteristics that are important for gesture analysis. Therefore, this paper investigates methods of temporal representation and variations of the Support Vector machines that consider temporal reasoning. Several experiments were performed in this context for gesture units segmentation. The best results were obtained with traditional Support Vector Machines applied to windowed data. In addition, three strategies of multiclass classification were applied to the problem of gesture phase segmentation. The results indicate that a good performance for gesture segmentation can be obtained by training the strategy with an initial part of the video to get an automated segmentation of the rest of the video. Thus, researchers in "gesture studies" could manually segment only part of the video, reducing the time needed to perform the analysis of gestures contained in long recordings.

Keywords: Gesture analysis, gesture unit segmentation, gesture phases segmentation, Support Vector Machines.

Sumário

Lista de Figuras **ix**

Lista de Tabelas **x**

Lista de Símbolos **xiv**

Lista de Abreviações **xvi**

1 Introdução **1**

1.1 Contextualização 2

1.2 Definição do Problema 5

1.3 Objetivos 6

1.4 Metodologia 7

1.5 Organização da Dissertação 9

2 Aprendizado de Máquina com Máquinas de Vetores Suporte **10**

2.1 Máquinas de Vetores Suporte 11

2.2 Incorporando Raciocínio Temporal 15

2.2.1 SVM com *Kernel* Recursivo 16

2.2.2 *Support Vector Echo-State Machines* 17

2.3 Considerações Finais 18

3 Representação Temporal **20**

3.1 Reconstrução do Espaço de Fases 20

3.2 Extração de Características no Domínio do Tempo 22

3.3 Características no Domínio da Frequência: Análise de Fourier 23

3.4 Características no Domínio Tempo-Frequência: Análise Wavelet 26

3.5 Considerações Finais 29

4 Análise Automatizada de Gestos 31

4.1 Estudos dos gestos e Análise Psicolinguística 32

4.1.1 Análise de gesticulação 34

4.1.2 Análise Psicolinguística 36

4.2 Segmentação das Fases dos Gestos 37

4.3 Considerações Finais 39

5 Experimentos, Resultados e Discussão 40

5.1 Conjuntos de Dados 41

5.2 Descrição dos Experimentos 45

5.3 Resultados 46

5.3.1 Experimento #1 – SVM tradicional com dados janelados 47

5.3.1.1 Etapa #1 48

5.3.1.2 Etapa #2 50

5.3.1.3 Etapa #3 50

5.3.2 Experimento #2 – SVM tradicional com características derivadas a partir dos dados janelados 52

5.3.2.1 Características no Domínio do Tempo 52

5.3.2.2 Características no Domínio da Frequência – Fourier 52

5.3.2.3 Características no Domínio Tempo-Frequência – Wavelet . 53

5.3.3 Experimento #3 – Reconstrução do Espaço de Fases 54

5.3.4 Experimento #4 – *Kernel* Recursivo 54

5.3.5 Experimento #5 – *Support Vector Echo-State Machine* 56

5.3.6 Experimento #6 – Estratégia um contra um 57

5.3.7 Experimento #7 – Estratégia um contra todos 58

5.3.8 Experimento #8 – Estratégia hierárquica 60

5.4 Considerações Finais 62

6 Estudo da dependência de usuário e de sessão 65

6.1 Experimento #9: usuário igual, sessão diferente 65

6.1.1 Segmentação da Unidade Gestual 66

6.1.2 Segmentação das Fases 67

6.2 Experimento #10: história igual, usuário diferente 69

6.2.1 Segmentação da Unidade Gestual 69

6.2.2 Segmentação das Fases 71

6.3 Experimento #11: misturando usuários, sessões e histórias 72

6.3.1 Segmentação da Unidade Gestual 72

6.3.2 Segmentação das Fases 74

6.4 Experimento #12: treinando e testando com o mesmo vídeo 76

6.4.1 Segmentação da Unidade Gestual 77

6.4.2 Segmentação das Fases 78

6.5 Considerações Finais 80

7 Conclusões 81

7.1 Principais Contribuições 83

7.2 Trabalhos Futuros 83

7.3 Considerações Finais 86

Referências 87

Lista de Figuras

1 Exemplo do gesto "balançar". 3

2 Sequência de *frames* anteriores ao *frame* de preparação. 4

3 Funcionamento da SVM 11

4 Função de decisão para três dados em $\Re^2$. 12

5 Funcionamento da SVESM 19

6 Reconstrução do Espaço de Fases 21

7 Sinal contínuo amostrado a uma frequência de 20Hz. 25

8 Amplitude e fase dos componentes sinusoidais do sinal original. 26

9 Representação da decomposição usando a transformada wavelet. 27

10 Sinal original e sinais reconstruídos a partir dos coeficientes de aproximação e detalhe, com as respectivas frequências obtidas para cada sinal usando transformada de Fourier. 29

11 Esquema gráfico ilustrando os procedimentos para segmentação das unidades gestuais 40

12 Sinal correspondente à velocidade escalar da mão direita. 44

13 Sinal correspondente à aceleração escalar da mão direita. 44

14 Erros dos melhores modelos para cada tamanho de recursão. 55

15 Sinal da velocidade da mão direita do vídeo A1. 67

16 Sinal da velocidade da mão direita do vídeo A3. 67

Lista de Tabelas

1 Informações sobre os vídeos que compõem o conjunto de dados. 41

2 Discordância apresentada por dois analistas ao rotular cada um dos vídeos do conjunto de dados (em %). 42

3 Descrição das etapas do experimento # 1. 48

4 Etapa #1 do experimento #1 com conjunto de treinamento não balanceado e balanceado, considerando deslocamento, tipo de característica, características e pontos de interesse – *F-score* dos modelos gerados. 49

5 Etapa #2 do experimento #1, considerando diferentes características e pontos de articulação – *F-score* dos modelos gerados. 50

6 *F-score* dos melhores modelos para cada característica no domínio do tempo. 52

7 *F-score* dos melhores modelos considerando análise de Fourier simples e n-dimensional. 53

8 *F-score* dos melhores modelos considerando análise Wavelet. 53

9 Desempenho dos melhores modelos gerados com Reconstrução do Espaço de Fases. 54

10 Desempenho dos melhores modelos gerados com *kernel* recursivo: erro e *F-score*. 55

11 Desempenho dos melhores modelos gerados com *kernel* recursivo: erro e *F-score*. 56

12 Desempenho do melhor conjunto de classificadores usando a estratégia um contra um – precisão (em %), revocação (em %) e *F-score*. 57

13 Desempenho do melhor conjunto de classificadores usando a estratégia um contra um – matriz de confusão. 58

14 Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – precisão (em %), revocação (em %) e *F-score*. 59

15 Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – matriz de confusão. 59

16 Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e *F-score*. 60

17 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – precisão (em %), revocação (em %) e *F-score*. 61

18 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão. 61

19 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão. 62

20 Desempenho do modelo para classificação entre descanso e gesto: treino com A1 e validação com A3. 66

21 Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do experimento #9. 66

22 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – precisão (em %), revocação (em %) e *F-score*. 68

23 Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e *F-score*. 68

24 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão. 69

25 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão. 69

26 Desempenho do modelo para classificação entre descanso e gesto: treino com A1 e validação com B1. 70

27 Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do experimento #10. 70

28 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão. 71

29 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – precisão (em %), revocação (em %) e *F-score*. 71

30 Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e *F-score*. 72

31 Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão. 72

32 Desempenho do modelo para classificação entre descanso e gesto: treino com A1+B1 e validação com A3+B3. 73

33 Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do vídeo A3 no experimento #11. 73

34 Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do vídeo B3 no experimento #11. 73

35 Desempenho do primeiro conjunto de classificadores para os vídeos A3 e B3 usando a estratégia hierárquica – precisão (em %), revocação (em %) e *F-score*. 75

36 Desempenho do primeiro conjunto de classificadores para o vídeo A3 usando a estratégia hierárquica – matriz de confusão. 75

37 Desempenho do primeiro conjunto de classificadores para o vídeo B3 usando a estratégia hierárquica – matriz de confusão. 75

38 Desempenho de cada classificador do primeiro conjunto analisado isoladamente para o vídeo A3 – precisão (em %), revocação (em %) e *F-score*. . . 76

39 Desempenho do primeiro conjunto de classificadores para o vídeo A3 usando a estratégia hierárquica – matriz de confusão. 76

40 Desempenho do primeiro conjunto de classificadores para o vídeo B3 usando a estratégia hierárquica – matriz de confusão dos erros por segmento. . . . 76

41 Desempenho do modelo para classificação entre descanso e gesto: treino com A1 (30%) e validação com A1 (70%). 77

42 Erros do modelo para os segmentos de descanso e gesto: treino com A1 (30%) e validação com A1 (70%). 77

43 Desempenho do modelo para classificação entre descanso e gesto: treino com A1 (70%) e validação com A1 (30%). 78

44 Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – precisão (em %), revocação (em %) e *F-score*. 78

45 Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e *F-score*. 79

46 Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – matriz de confusão. 79

47 Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – matriz de confusão dos erros por segmento. 79

48 Erros do modelo para os segmentos de fases: treino com A1 (70%) e validação com A1 (30%). 80

Lista de Símbolos

$\boldsymbol{u_i}$ i-ésima entrada da Máquina de Vetores Suporte.

$\boldsymbol{y_i}$ Atributo classe da i-ésima entrada u, ou saída da Máquina de Vetores Suporte para a entrada u.

$\boldsymbol{w}$ Conjunto de pesos da Máquina de Vetores Suporte.

$\boldsymbol{b}$ Bias da Máquina de Vetores Suporte.

$\boldsymbol{C}$ Termo de regularização que visa balancear a minimização do erro empírico e a minimização da complexidade do modelo.

$\boldsymbol{\xi_i}$ Erro referente à i-ésima entrada em relação à saída obtida pela Máquina de Vetores Suporte.

$\boldsymbol{\varphi(u_i)}$ Mapeamento sobre a i-ésima entrada da Máquina de Vetores Suporte.

$\boldsymbol{\alpha_i}$ Multiplicador de Lagrange referente ao dado u_i.

$\boldsymbol{K(u_i, u_j)}$ Função *kernel* aplicado aos dados u_i e u_j, equivalente ao produto interno $\langle \varphi(u_i) \cdot \varphi(u_j) \rangle$.

$\boldsymbol{\sigma}$ Parâmetro da função *kernel*.

$\boldsymbol{s_i}$ i-ésima série temporal, considerada como entrada do *kernel* recursivo e da *Support Vector Echo-State Machine*; uma observação no tempo t da série é referenciada como $s_i(t)$.

$\boldsymbol{\kappa_t(s_i, s_j)}$ Função *kernel* recursiva aplicada às séries temporais s_i e s_j no tempo t.

$\boldsymbol{\sigma_h}$ Parâmetro de funções *kernel* recursivas, referentes ao histórico da série.

$\boldsymbol{\sigma_i}$ Parâmetro de funções *kernel* recursivas, referentes ao i-ésimo dado da série.

$\boldsymbol{\eta}$ Parâmetro de funções *kernel* recursivas que define o tamanho da recursão.

$\boldsymbol{x_i}$ Estado do reservatório referentes ao i-ésimo dado da série considerado na *Support Vector Echo-State Machine*.

$\boldsymbol{\tau}$ Parâmetro de atraso considerado na Reconstrução do Espaço de Fases.

$\boldsymbol{m}$ Parâmetro de dimensão incorporada considerado na Reconstrução do Espaço de Fases.

$\boldsymbol{f_s}$ Frequência de amostragem do sinal, utilizada para realizar a discretização de um sinal contínuo antes de aplicar a transformada de Fourier discreta.

$\boldsymbol{G(z)}$ Filtro passa-baixa usado para realizar a transformada Wavelet.

$\boldsymbol{H(z)}$ Filtro passa-alta usado para realizar a transformada Wavelet.

$\boldsymbol{cA}$ Coeficientes de aproximação obtidos pela transformada Wavelet.

$\boldsymbol{cD}$ Coeficientes de detalhe obtidos pela transformada Wavelet.

$\boldsymbol{p}$ Posição de um ponto de articulação, dada por um vetor de coordenadas nos eixos x, y e z.

$\boldsymbol{p_c}$ Posição da cabeça, dada por um vetor de coordenadas nos eixos x, y e z.

$\boldsymbol{p_t}$ Posição do tronco, dada por um vetor de coordenadas nos eixos x, y e z.

$\boldsymbol{p_{norm}}$ Posição normalizada de um ponto de articulação a partir da posição p, dada por um vetor de coordenadas nos eixos x, y e z.

$\boldsymbol{v_{vet}}$ Velocidade vetorial de um ponto de articulação.

$\boldsymbol{a_{vet}}$ Aceleração vetorial de um ponto de articulação.

Lista de Abreviações

SVM Máquina de Vetores Suporte – *Support Vector Machine.*

VC Vapnik-Chervonenkis.

SVR Regressão por Vetores Suporte – *Support Vector Regression.*

RBF Função de Base Radial – *Radial Basis Function.*

SVESM Máquina de Vetores Suporte com Estados de Eco – *Support Vector Echo State Machine.*

SVA Soma dos Valores Absolutos.

MAV Valor Médio Absoluto – *Mean Absolute Value.*

MMAV Valor Médio Absoluto ponderado por uma função janela – *Modified Mean Absolute Value.*

RMS Raiz Quadrada Média – *Root Mean Squared.*

SSI Integral Quadrada Simples – *Simple Squared Integral.*

VAR Variância – *Variance.*

DFT Transformada de Fourier Discreta – *Discrete Fourier Transform.*

DWT Transformada Wavelet Discreta – *Discrete Wavelet Transform.*

HMM Modelo Oculto de Markov – *Hidden Markov Model.*

CMF Fusão de Modalidade Condicional – *Conditional Modality Fusion.*

ROC Característica de Operação do Receptor – *Receiver Operating Characteristic.*

EM Esperança-Maximização – *Expectation-Maximization.*

DBN Rede Bayesiana Dinâmica – *Dynamic Bayesian Network.*

WFT Transformada de Fourier Janelada – *Windowed Fourier Transform.*

CWT Transformada Wavelet Contínua – *Continuous Wavelet Transform.*

FSM Máquina de Estados Finitos – *Finite State Machine.*

1 Introdução

Nos últimos anos, tem sido possível notar uma quantidade significativa de pesquisas relacionadas à análise de gestos na computação. Esse tipo de pesquisa geralmente visa o desenvolvimento de métodos cada vez mais naturais de interação humano computador ou mais pervasivos ao comportamento humano, usando gestos, por exemplo, como meio de ativação de recursos de *hardware* ou *software*. Recentemente, houve um crescimento no número de pesquisas em análise de gestos que visam dar suporte à interpretação do papel da gestualidade na constituição do discurso, permitindo assim estabelecer relações que melhorem a compreensão do que está sendo falado. Tais pesquisas são baseadas na área de "estudos dos gestos" – área interdisciplinar que visa estudar o uso das mãos e outras partes do corpo para fins comunicativos, combinando conhecimentos de linguística, psicologia, ciências sociais e outras áreas (KENDON, 1996).

Pesquisadores da área de estudos dos gestos geralmente gravam vídeos de pessoas falando ou gesticulando em língua de sinais – contando histórias, discursando ou conversando – e, a partir das gravações, analisam os gestos realizados por elas. Tal análise exige a transcrição de componentes dos vídeos, como fala, tom de voz e gestos. No caso da transcrição de gestos, interesse do presente trabalho, uma das etapas consiste na segmentação do gesto de acordo com fases básicas – preparação, *stroke*, *hold* e retração (KENDON, 1980). Essa segmentação é normalmente realizada de forma manual pelos pesquisadores, consistindo em um processo de alto custo em relação a tempo e esforço. Assim, o oferecimento de processos automatizados que auxiliam na etapa de análise dos gestos, anterior à interpretação dos gestos em si, é de grande ajuda para tais pesquisadores. De fato, Vinciarelli et al. (2008) defende que a criação de ferramentas computacionais tem o potencial de revolucionar a pesquisa básica em ciências sociais, possibilitando um aumento da qualidade da pesquisa e uma redução do tempo necessário para conduzi-la. Esta pesquisa atualmente é, de forma geral, lenta, trabalhosa e frequentemente imprecisa.

Neste trabalho, pretende-se estudar algoritmos de classificação supervisionada para

realizar a segmentação das fases dos gestos manuais[1]. Dentre os algoritmos existentes para esse tipo de tarefa, optou-se por explorar as Máquinas de Vetores Suporte (SVM, do inglês *Support Vector Machines*) (VAPNIK, 1998). Esse método tem sido amplamente utilizado em diversas áreas de aplicação, incluindo aplicações de análise de movimentos e gestos (LEE; GRIMSON, 2002; BEGG; PALANISWAMI, 2005; SCHULDT et al., 2004). Porém, os trabalhos nessa área geralmente lidam com dados de movimentos e gestos representando o tempo através de uma sequência de características espaciais ou cinemáticas, e não consideram a dependência temporal existente nos dados como informação intrínseca ao raciocínio automatizado aplicado nas tarefas de análise.

Assim, neste contexto, no presente trabalho foram consideradas abordagens híbridas que combinem SVM e técnicas de análise de dados que considerem a dependência temporal dos dados, visando aplicar representação e raciocínio temporal em algoritmos para segmentação das fases dos gestos. A hipótese considerada neste estudo é que abordagens híbridas combinando aprendizado de máquina utilizando SVM e representação ou raciocínio temporal sejam adequadas à análise temporal de gestos, e têm potencial para suportar a criação de ferramentas úteis para a área de análise de gestos.

1.1 Contextualização

Neste trabalho, considera-se a hierarquia proposta por Kita et al. (1998) para descrever as fases dos gestos. Essa hierarquia é baseada na hierarquia de Kendon (1980) e organiza os gestos a partir de dois conceitos importantes: unidade gestual (*gesture unit* ou *G-unit*), que é o período de tempo que começa quando o braço se afasta de uma posição de descanso e termina quando o braço retorna à posição de descanso; e a frase gestual (*gesture phrase* ou *G-phrase*), que é um segmento dentro da unidade gestual composto por uma ou mais fases que compreende o significado do gesto.

Cinco fases são definidas por Kendon (1980) e McNeill (1992):

- preparação: as mãos se movem para a posição na qual o movimento começa;
- *hold* pré-*stroke*: pequena pausa ao final da preparação, mantendo a posição e configuração das mãos;
- *stroke*: movimento que expressa o significado do gesto – é a única fase obrigatória no gesto, as demais são opcionais;

[1]Por questões de simplificação, gestos manuais serão tratados apenas como "gestos" neste trabalho.

- *hold* pós-*stroke*: pequena pausa ao final do *stroke*, mantendo a posição e configuração das mãos;
- retração: retorno das mãos à posição de descanso.

Kita et al. (1998) amplia a hierarquia, propondo a fase de *hold* independente, que ocorre quando o significado do gesto é expressado por uma pausa da mão em determinada posição – ou seja, um *hold* – e não há *stroke* associado. Neste caso, o *hold* independente substitui o *stroke* na expressão de uma mensagem.

A Figura 1 ilustra as fases de um gesto com o significado "balançar": o primeiro quadro mostra o usuário em posição de descanso; o segundo quadro representa o momento da fase de preparação, na qual a mão se desloca da posição de descanso para o lugar onde o *stroke* é realizado; o terceiro quadro mostra um *frame* do *stroke*; e o quarto quadro mostra um instante do retorno do usuário à posição de descanso.

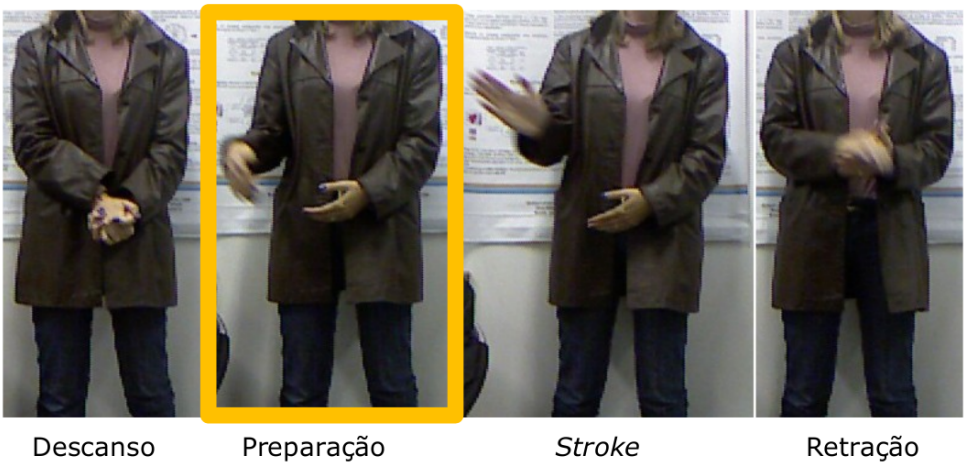

Figura 1 – *Exemplo do gesto "balançar": o primeiro frame pertence à posição de descanso, o segundo pertence à fase de preparação, o terceiro ao stroke e o quarto à fase de retração.*

Uma questão relevante relacionada à segmentação das fases do gesto é a dependência dos *frames* anteriores e posteriores. A Figura 2 mostra o *frame* correspondente à preparação na Figura 1 e os *frames* anteriores para mostrar que a segmentação das fases não pode ser realizada apenas a partir da imagem do *frame* de interesse. Observando apenas a imagem, não é possível definir a fase do gesto. Porém, ao observar que as mãos do usuário estavam "repousando" durante uma determinada quantidade de *frames* e depois se movem até determinado local para começar um *stroke*, é possível dizer que os *frames* desse movimento correspondem a um período de preparação.

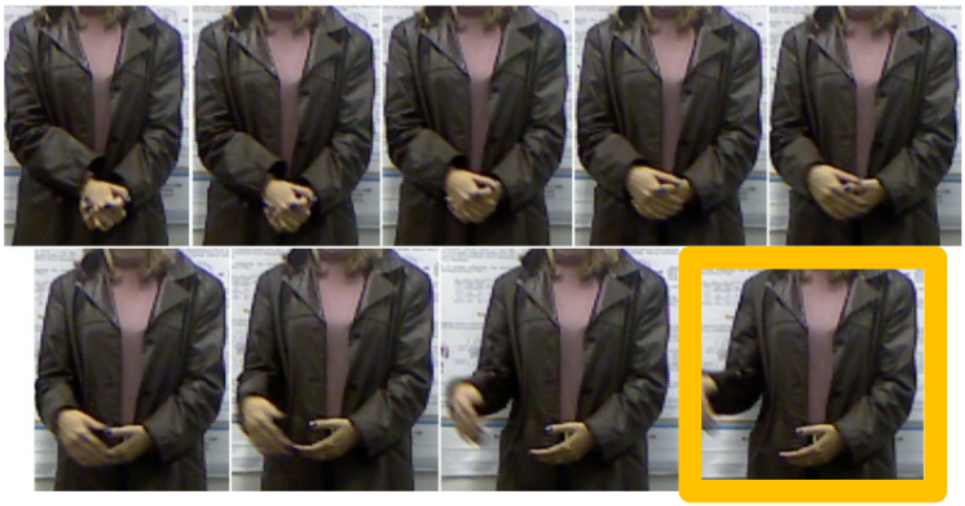

Figura 2 – *Sequência de frames anteriores ao frame de preparação da Figura 1, mostrando que é necessário considerar uma sequência de frames para definir a fase de um frame.*

Também é importante notar que algumas fases podem apresentar comportamentos semelhantes. Posições de descanso e períodos de *hold* são períodos nos quais as mãos ficam paradas, apresentando padrões de velocidade similares. A diferença entre tais períodos poderia ser mais facilmente notada se fossem consideradas: posição da pausa, visto que as posições de descanso ocorrem em locais recorrentes; tensão das mãos, visto que a mão geralmente está mais relaxada nas posições de descanso e mais tensa nos períodos de *hold*; e fases anteriores e posteriores, já que o descanso é precedido pela retração e seguido pela preparação, enquanto períodos de *hold* podem ser precedidos por fases de descanso, preparação e *stroke*, e seguidos por fases de *stroke*, retração e descanso.

As fases de preparação e retração também podem apresentar perfis semelhantes, visto que essas fases visam, respectivamente, posicionar a mão para a realização da fase expressiva do gesto e retornar da fase expressiva à posição de descanso.

Além da complexidade envolvida na distinção entre determinadas fases do gesto, ainda é preciso considerar que a tarefa de segmentação tem caráter subjetivo. Mesmo entre analistas humanos, existe um alto grau de divergência nas segmentações realizadas por eles, diante de um mesmo gesto. Inclusive, um mesmo analista pode gerar segmentações diferentes para um mesmo vídeo se realizar a análise dos mesmos gestos em dias, ou momentos, diferentes.

Além disso, a segmentação das fases na área de estudos dos gestos ainda é um problema de pesquisa com diversas questões em aberto, como:

- Deve-se considerar as mãos de forma individual?
- Deve-se considerar mudanças nas posições de descanso como uma especificidade do descanso? Essas mudanças carregam algum significado?
- Como considerar gestos sem sentido semântico bem definido, como "arrumar os óculos"ou "coçar a cabeça"?

Portanto, é pertinente dizer que o problema de segmentação das fases dos gestos é um problema de reconhecimento de padrões complexo, apresentando elementos de dificuldade relacionados à subjetividade, classes similares e falta de uma teoria formalmente definida e amplamente aceita sobre muitos aspectos relacionados à tomada de decisão sobre a tarefa de segmentação.

1.2 Definição do Problema

Neste trabalho, um vídeo é uma sequência de *frames* $S = \{f_1, f_2, \cdots, f_n\}$ de tamanho n, e é utilizado como entrada para um modelo classificador com o objetivo de segmentar as fases dos gestos. A sequência de *frames* consiste, fisicamente, em uma sequência de imagens estáticas em RBG. Neste trabalho, as sequências de *frames* contêm imagens de uma pessoa contando uma história e o problema de segmentação das fases dos gestos consiste em um problema de classificação. A entrada para o problema de classificação é um *frame* f_i da sequência S e deve-se classificá-lo de acordo com uma das classes $c_i = \{D, P, S, H, R\}$, que correspondem, respectivamente, às fases de **d**escanso, **p**reparação, ***s**troke*, ***h**old* e **r**etração.

O problema de segmentação das fases dos gestos pode ser dividido em subtarefas de classificação mais simples. Devido à complexidade do problema, é útil criar subtarefas de classificação binária, facilitando o processo de classificação. A simplificação do problema através da sua divisão em subtarefas binárias também é adequada à SVM, visto que a SVM tradicional foi desenvolvida para problemas binários.

Existem diferentes formas de dividir uma tarefa em subtarefas de classificação binárias: (a) usando a estratégia um contra um, na qual cria-se um classificador binário para cada dupla de classes do problema e a resposta final da estratégia é escolhida por votação; (b) usando a estratégia um contra todos, na qual cria-se um classificador binário para distinguir cada classe do problema de todas as demais classes do problema, e a resposta

final da estratégia é dada pela classe com maior saída da SVM; e (c) usando uma estratégia hierárquica, como a descrita a seguir:

1. Classificação de posições de descanso: dada uma entrada $f_i \in S$, a saída é $c_i = \{D, U\}$, onde $U \supset \{P, S, H, R\}$ e corresponde ao conceito de unidade gestual;

2. Classificação de *hold*: dada uma entrada $f_i \in S_U$, onde S_U é o subconjunto de *frames* de S classificado como pertencente à classe U pelo classificador de posições de descanso, a saída é $c_i = \{H, M\}$, onde $M \supset \{P, S, R\}$ e corresponde às fases que contém maior quantidade de movimento dentro da unidade gestual;

3. Classificação de *stroke*: dada uma entrada $f_i \in S_M$, onde S_M é o subconjunto de *frames* de S_U classificado como pertencente à classe M pelo classificador de *hold*, a saída é $c_i = \{S, T\}$, onde $T \supset \{P, R\}$ e corresponde às demais fases da unidade gestual;

4. Classificação de fase de preparação/retração: dada uma entrada $f_i \in S_T$, onde S_T é o subconjunto de *frames* de S_M classificado como pertencente à classe T pelo classificador de *stroke*, a saída é $c_i = \{P, R\}$.

1.3 Objetivos

O objetivo geral deste trabalho é desenvolver estratégias para segmentação automatizada das unidades gestuais e das fases dos gestos contidos em um vídeo, capazes de suportar a interpretação do papel da gestualidade na constituição do discurso. Para atingir esse objetivo, o problema de segmentação das fases dos gestos foi formulado como uma tarefa de classificação supervisionada. A SVM foi escolhida como método de classificação devido aos bons resultados que tem apresentado para diversos problemas e à sua capacidade de generalização. Procurando melhor adequação da SVM ao problema de análise de gestos, abordagens que consideram a dependência temporal dos dados foram investigadas.

Nesse contexto, os seguintes objetivos específicos foram estabelecidos:

- Estudo do estado da arte de abordagens híbridas, combinando SVM e processamento temporal.

- Levantamento do estado da arte da área de aplicação – análise de gestos.

- Avaliação da adequabilidade dessas abordagens híbridas no tratamento de problemas de análise de gestos.
- Modelagem de algoritmos de aprendizado de máquina para suportar a análise proposta.
- Análise dos resultados obtidos pelas estratégias propostas.

1.4 Metodologia

Esta seção visa descrever, em linhas gerais, a metodologia utilizada neste trabalho a fim de atingir os objetivos propostos. Com exceção da estratégia de revisão sistemática, todas as outras estratégias de ação são descritas em mais detalhes nos Capítulos 5 e 6, onde elas são aplicadas.

Inicialmente, para realizar a revisão de conceitos fundamentais de SVM e sobre a área de estudos de gestos, foram feitas revisões bibliográficas baseadas na leitura de livros e artigos clássicos de cada área, recomendados pela orientadora e por professores das áreas relacionadas.

Já para realizar o levantamento do estado da arte, foi utilizada a metodologia de revisão sistemática proposta por Kitchenham (2004). Segundo essa autora, a revisão sistemática é uma metodologia para levantamento bibliográfico que consiste em identificar, avaliar e interpretar as pesquisas relevantes dentro de uma determinada área de pesquisa. O objetivo desta metodologia é criar uma organização sistemática para conduzir a revisão de forma que seja possível documentar todo o processo de levantamento bibliográfico, possibilitando aos leitores a oportunidade de avaliar a qualidade da revisão ou até mesmo reproduzi-la. A fim de cobrir os conceitos envolvidos com o estudo proposto neste projeto de mestrado, duas revisões sistemáticas foram executadas: a) estado da arte de abordagens híbridas, que combinem SVM e processamento temporal; b) estado da arte sobre análise de gestos, focando em aspectos dinâmicos e aplicações nas áreas de análise do discurso e conversação natural.

Para a fase de coleta de dados, o dispositivo Microsoft Kinect foi utilizado. Foram gravados vídeos de dois usuários contando três histórias diferentes, motivadas pela leitura de três histórias em quadrinhos. O dispositivo foi usado para captar, para cada *frame*, as posições (coordenadas x, y e z) de pontos de articulação e a imagem correspondente àquele *frame*. A partir da análise das imagens, cada vídeo foi rotulado com dois objetivos: a) criar

um conjunto de dados adequado para a resolução da tarefa de classificação; b) avaliar e mensurar a ocorrência de discordância de rotulação em relação à segmentação de um mesmo vídeo, visto que a segmentação de gestos é uma tarefa de caráter subjetivo. Para esse último objetivo, foram consideradas rotulações realizadas por dois analistas.

Na implementação dos experimentos, optou-se por utilizar a ferramenta Matlab com a toolbox Spider. Todos os testes do modelo de classificação foram feitos com a técnica *holdout*, na qual parte do conjunto de dados é usada para treinamento e outra parte é usada para validação. Cada uma das divisões do conjunto (a parte para treino e a parte para validação) variou de acordo com o objetivo do experimento. Os primeiros experimentos foram realizados para avaliar a adequabilidade das Máquinas de Vetores Suporte com representação temporal e raciocínio temporal ao problema de segmentação das unidades gestuais. Os experimentos posteriores avaliaram três estratégias de classificação multiclasse – um contra um, um contra todos e hierárquica – aplicadas ao problema de segmentação das fases dos gestos. Por fim, foram realizados experimentos para avaliar as melhores estratégias em relação à dependência de usuário, sessão e história.

As estratégias obtidas foram analisadas sob duas perspectivas: a perspectiva de aprendizado de máquina, visando aferir a qualidade dos modelos classificadores; e a perspectiva da área de aplicação, com o objetivo de avaliar a adequabilidade dos modelos às necessidades da área de estudos dos gestos e de comparar os resultados obtidos com os trabalhos correlatos.

Sob a perspectiva de aprendizado de máquina, os modelos classificadores foram avaliados considerando as métricas de erro de classificação, precisão, revocação, *F-score* e matrizes de confusão, avaliando erros *frame* a *frame*.

Sob a perspectiva da área de aplicação, foram analisados os segmentos correspondentes a cada fase. Nesse caso, foram analisados erros por segmento, visando possibilitar alguma forma de comparação em relação aos trabalhos correlatos. Matrizes de confusão foram usadas para auxiliar na análise dos erros por segmento. A partir da análise dos segmentos, também foram analisados erros por *frame* a fim de investigar as causas do erro: por exemplo, erros no início e no fim de segmentos – chamados erros de transição – são comuns e não são tão prejudiciais sob a ótica da área de aplicação quanto outros tipos de erro.

1.5 Organização da Dissertação

Esta dissertação está dividida em 7 capítulos, considerando esta introdução. Os demais capítulos estão organizados da seguinte forma:

- O Capítulo 2 contém um resumo sobre a teoria de SVM, necessária para a compreensão da estratégia de classificação aplicada neste trabalho, e apresenta a teoria sobre *kernel* recursivo e *Support Vector Echo-State Machines*, que são variações que incorporam raciocínio temporal à SVM.

- O Capítulo 3 reúne algumas formas de pré-processamento temporal dos dados, descrevendo, em linhas gerais, o método de Reconstrução do Espaço de Fases, alguns métodos para extração de características no domínio do tempo, a transformada de Fourier e a transformada Wavelet.

- O Capítulo 4 apresenta a revisão da literatura correlata à área de análise de gestos. Essa revisão diz respeito principalmente a aspectos da teoria de gestos e traz uma série de trabalhos relacionados à aplicação em análise de gesticulação e à segmentação das fases dos gestos em si.

- O Capítulo 5 descreve os conjuntos de dados utilizados para os experimentos de avaliação da aplicação de SVM no problema de segmentação das fases dos gestos, considerando o uso de representação temporal e o uso de raciocínio temporal; também detalha a forma como os experimentos foram organizados, os resultados obtidos e algumas análises sobre esses resultados.

- O Capítulo 6 apresenta alguns experimentos adicionais que exploram variações no contexto de aplicação dos modelos classificadores e das abordagens para segmentação de todas as fases do gesto. Tais experimentos visam analisar a dependência de usuário, sessão e história no problema de segmentação das fases dos gestos.

- Por fim, no Capítulo 7 são apresentadas as considerações finais deste trabalho, com destaque para as principais contribuições do estudo e as possibilidades de extensão do mesmo.

2 Aprendizado de Máquina com Máquinas de Vetores Suporte

Aprendizado de máquina é definido por Mitchell (1997) como a capacidade de um programa de computador melhorar seu desempenho em relação a determinada tarefa a partir de experiências anteriores. Tal definição diz respeito ao aprendizado indutivo, no qual o sistema tenta induzir uma regra geral a partir de instâncias já observadas sobre um fenômeno. Formalmente, Mitchell (1997) apresenta a seguinte definição sobre aprendizado indutivo: "Um programa de computador *aprende*, a partir de experiências E com respeito a uma classe de tarefas T e medida de desempenho M, se seu desempenho na tarefa T, conforme medido por M, melhora com as experiências E".

No caso de tarefas de classificação, o conjunto de experiências apresentado ao programa é chamado de conjunto de treinamento e consiste em um conjunto com N amostras, definido por $\{\boldsymbol{u_i}, y_i\}_{i=1}^{N}$, onde $\boldsymbol{u_i} \in \Re^A$ é a entrada do modelo com A atributos e y_i é a saída, o atributo *classe* que deve ser aprendido pelo programa. O objetivo é que, após uma fase de aprendizado (ou treinamento), o programa seja capaz de receber uma nova instância, sem o atributo classe, e retornar a classe correta associada àquela instância.

Diversas abordagens são utilizadas para realizar aprendizado indutivo, como Redes Neurais, Árvores de Decisão e Máquinas de Vetores Suporte (*Support Vector Machines* – SVM). Neste trabalho, foca-se na SVM, uma abordagem inicialmente proposta para problemas de classificação binária que visa, por meio de exemplos, aproximar uma superfície de decisão entre as classes. A principal vantagem da SVM em relação às demais alternativas é que a SVM visa a minimização do risco estrutural. Ou seja, enquanto as demais abordagens consideram apenas o risco empírico calculado a partir do conjunto de treinamento, a SVM possibilita gerar uma superfície de decisão com maior capacidade de generalização, evitando o sobreajuste aos dados de treinamento (HAYKIN, 2008).

Neste trabalho, pretende-se explorar a SVM como método de aprendizado indutivo para classificar padrões temporais. Porém, como a maior parte dos métodos de apren-

dizado indutivo, a SVM tradicional considera os exemplos como independentes e identicamente distribuídos e, portanto, não considera as dependências temporais existentes entre as instâncias. Assim, este trabalho explora duas alternativas para considerar tais dependências temporais: usar algum tipo de pré-processamento temporal dos dados ou modelagem temporal; ou modificar o funcionamento da SVM para incorporar raciocínio temporal ao modelo.

Este capítulo apresenta os conceitos fundamentais sobre SVM (Seção 2.1) e duas técnicas que incorporam raciocínio temporal à SVM (Seção 2.2). Métodos para realizar pré-processamento temporal são tratados separadamente no Capítulo 3.

2.1 Máquinas de Vetores Suporte

As SVMs são baseadas na realização de um mapeamento não linear[1] dos vetores de entrada do seu espaço de características (espaço original dos dados sob análise) para um espaço de características de alta dimensão; e na aproximação de uma superfície de decisão, considerando sua organização em classes, nesse novo espaço de características (HAYKIN, 2008). Esse processo é ilustrado na Figura 3.

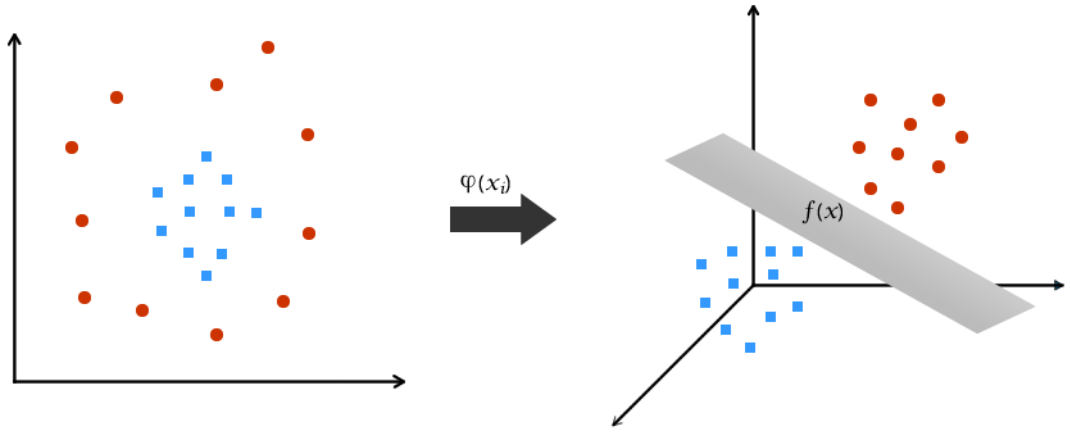

Figura 3 – *Funcionamento da SVM: os dados do espaço original são mapeados para um espaço de características de alta dimensão; nesse novo espaço, aproxima-se um hiperplano para separação dos dados (imagem adaptada de Lorena e Carvalho (2007)). Na figura, $\varphi(x_i)$ é o mapeamento aplicado aos vetores e $f(x)$ é a superfície de decisão entre as classes.*

Em princípio, a SVM tem o objetivo de maximizar a capacidade de generalização

[1] A SVM pode ser utilizada com mapeamento linear para problemas linearmente separáveis. No entanto, para problemas complexos, é comum o uso de mapeamentos não lineares que permitem transpor os dados para um espaço de características de alta dimensão, no qual é provável que os dados sejam linearmente separáveis. O foco deste trabalho é o modelo SVM não linear.

e evitar a ocorrência do fenômeno de sobreajuste na aproximação de uma superfície de decisão para um problema de aprendizado indutivo. Para isso, a SVM considera alguns conceitos da teoria de aprendizado estatístico – dimensão VC, risco estrutural e risco empírico.

A dimensão Vapnik-Chervonenkis (VC) mede a capacidade ou poder expressivo de uma família de funções de classificação representadas por uma máquina de aprendizado (HAYKIN, 2008). Conforme definido por Vapnik e Chervonenkis (1971), a dimensão VC de uma classe de funções é h se e somente se existe um conjunto de amostras $\{u_i\}_{i=1}^{h}$ tal que, para qualquer uma das 2^h configurações possíveis de rotulação binária, as amostras podem ser corretamente classificadas; e não existe nenhum conjunto $\{u_i\}_{i=1}^{q}$ com $q > h$ satisfazendo essa propriedade. Tal definição pode ser ilustrada pela Figura 4, que contém as possibilidades de rotulação de três amostras no $\Re^2$ e as possíveis classificações dadas por uma função linear. Neste caso, a dimensão VC é 3, visto que apenas três dados amostras ser classificadas corretamente para qualquer possibilidade de rotulação desses dados.

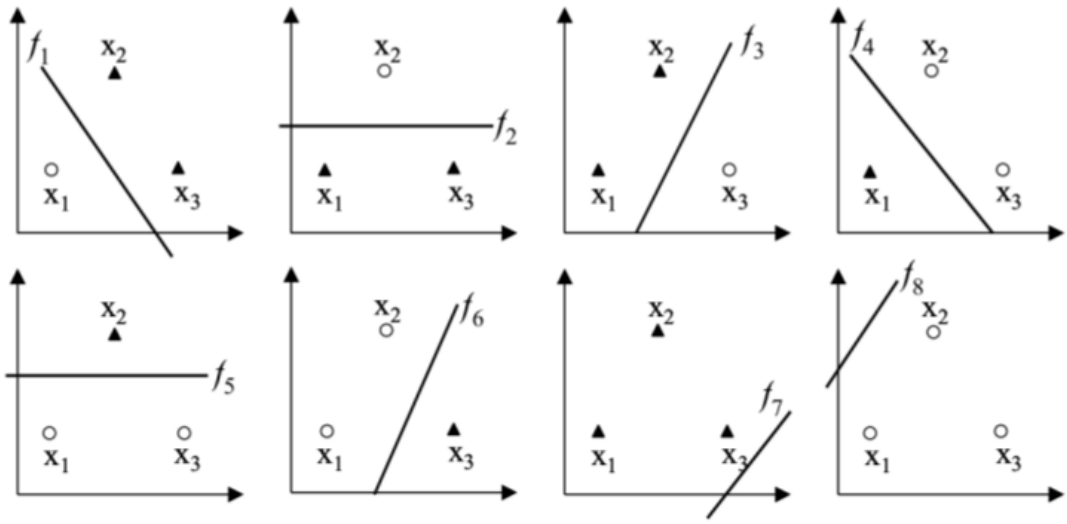

Figura 4 *– Três dados em $\Re^2$, com as classificações permitidas por uma função de decisão linear (LORENA; CARVALHO, 2007).*

O risco empírico corresponde ao erro do modelo obtido em relação às amostras do treinamento. Considerar apenas o risco empírico, porém, pode gerar modelos complexos, sobreajustados aos dados de treinamento e, portanto, com pouco poder de generalização. Uma das formas de evitar o sobreajuste do modelo aos dados do treinamento é controlar a dimensão VC do modelo, garantindo um modelo de baixa complexidade. Tal controle pode ser feito por meio da minimização do risco estrutural, o qual visa a escolha de uma dimensão VC apropriada para a resolução de um determinado problema. Assim, a SVM

trata o problema de encontrar um modelo que minimize o risco empírico e que pertença a uma classe de funções com baixa dimensão VC (LORENA; CARVALHO, 2007), como descrito a seguir.

Para entender o processo de aprendizado envolvido nesse contexto, considere uma SVM para resolução de um problema de classificação binário, que usa um conjunto de treinamento conforme definido anteriormente, porém com saída $y_i \in \{-1, +1\}$. A SVM objetiva encontrar um hiperplano ótimo para a separação das classes, dado por $f(\boldsymbol{u_i}) = \langle \boldsymbol{w_o} \cdot \varphi(\boldsymbol{u_i}) \rangle + b_o$, onde $\boldsymbol{w_o}$ é um conjunto ótimo de pesos, b_o é o *bias* ótimo, e φ é o mapeamento aplicado aos vetores de entrada.

O hiperplano ótimo pode ser obtido por:

$$\min \phi(\boldsymbol{w}) = \frac{1}{2}\langle \boldsymbol{w} \cdot \boldsymbol{w} \rangle + C \sum_{i=1}^{N} \xi_i, \tag{2.1}$$

onde $\boldsymbol{w}$ é um conjunto de pesos, b é o *bias*,C é um fator de regularização, e ξ é um fator de erro, sujeito a

$$\begin{aligned} & y_i(\langle \boldsymbol{w} \cdot \varphi(\boldsymbol{u_i}) \rangle + b) \geq 1 - \xi_i, \quad i = 1, ..., N, \\ & \xi_i \geq 0, \quad i = 1, \cdots, N. \end{aligned}$$

Observando a equação 2.1, é possível notar a minimização do risco empírico, representada pela minimização do segundo termo da equação. Já a minimização do risco estrutural pode ser alcançada por meio da minimização do primeiro termo da equação, conforme demonstrado por Haykin (2008).

O problema de otimização da equação 2.1 é um problema de programação quadrática, visto que a função objetivo é quadrática enquanto as restrições são lineares (CRISTIANINI; SHAWE-TAYLOR, 2000). Para resolver esse tipo de problema, é possível aplicar o método lagrangiano, que traduz o problema original, também chamado problema primal, em um novo problema, chamado problema dual, composto por uma função objetivo lagrangiana e uma combinação linear de restrições, de forma que os coeficientes dessa combinação são chamados multiplicadores de Lagrange. Aplicando o método ao problema de otimização da equação 2.1[2], obtém-se a função lagrangiana:

[2]Para verificar a derivação realizada utilizando o método de Langrange, veja Cristianini e Shawe-Taylor (2000).

$$\max \mathcal{L}_1(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \langle \varphi(\boldsymbol{u_i}) \cdot \varphi(\boldsymbol{u_j}) \rangle, \tag{2.2}$$

sujeito a

$$\sum_{i=1}^{N} \alpha_i y_i = 0,$$
$$C \geq \alpha_i \geq 0 \quad \text{for } i = 1, 2, \cdots, N,$$

onde $\boldsymbol{\alpha}$ são multiplicadores de Lagrange.

Resolvendo o problema na equação 2.2, é possível obter a solução para a equação 2.1, já que $\boldsymbol{w}$ pode ser definido em termos de $\boldsymbol{\alpha}$, como demonstrado em Haykin (2008). Na equação 2.2, uma função *kernel* $K(\boldsymbol{u_i}, \boldsymbol{u_j})$ pode ser usada para calcular o produto interno $\langle \varphi(\boldsymbol{u_i}) \cdot \varphi(\boldsymbol{u_j}) \rangle$ diretamente a partir dos dados de entrada originais. Assim, o mapeamento $\varphi(u)$ para um espaço de características de alta dimensão é realizado de forma implícita. Algumas das funções *kernel* comumente utilizadas (HAYKIN, 2008) incluem:

- *Kernel* linear:

$$K(u_i, u_j) = \langle u_i \cdot u_j \rangle.$$

- *Kernel* polinomial:

$$K(u_i, u_j) = \langle u_i \cdot u_j \rangle^d.$$

- *Kernel* gaussiano (também conhecido como *Radial Basis Function* – RBF):

$$K(u_i, u_j) = \exp\left(\frac{\|u_i - u_j\|^2}{2\sigma^2}\right).$$

Para aplicações de natureza temporal, como a aplicação estudada nesta dissertação, é interessante considerar também a tarefa de regressão. As SVMs podem ser adaptadas para resolver esse tipo de tarefa e são, nesse caso, chamadas de Support Vector Regression (SVR). A SVR considera as variáveis de folga ξ_i decompostas em ξ_i e $\hat{\xi}_i$, representando, respectivamente, erros acima ou abaixo da saída esperada. A formulação do problema de regressão depende da função de perda escolhida. Uma das funções de perda mais comuns é a função ϵ-insensível, que é formulada como:

$$L_\epsilon = \begin{cases} 0, & \text{se } |y_i - f(\boldsymbol{u_i})| \leq \epsilon \\ y_i - f(\boldsymbol{u_i}) - \epsilon, & \text{se } |y_i - f(\boldsymbol{u_i})| > \epsilon, \end{cases} \tag{2.3}$$

onde ϵ é um parâmetro definido pelo usuário que define o desvio máximo aceito pelo modelo, ou seja, erros menores que ϵ não são considerados.

Neste caso, o problema é formulado por meio da minimização de:

$$\min \phi(\boldsymbol{w}) = \frac{1}{2}\langle \boldsymbol{w} \cdot \boldsymbol{w} \rangle + C \sum_{i=1}^{N} \left(\xi_i + \hat{\xi}_i \right),$$

sujeito às restrições:

$$\begin{aligned} y_i - \langle \boldsymbol{w} \cdot \varphi(\boldsymbol{u_i}) \rangle - b &\leq \epsilon + \xi_i \\ \langle \boldsymbol{w} \cdot \varphi(\boldsymbol{u_i}) \rangle + b - y_i &\leq \epsilon + \hat{\xi}_i \\ \xi_i, \hat{\xi}_i &\geq 0. \end{aligned}$$

2.2 Incorporando Raciocínio Temporal

Conforme mencionado no início deste capítulo, a teoria por trás da SVM parte do pressuposto que os dados são independentes e identicamente distribuídos. No caso de dados com aspectos temporais, tal pressuposto não é verdadeiro, visto que existe dependência temporal entre as amostras. Assim, ao utilizar a SVM tradicional para processar dados temporais, despreza-se a característica temporal presente nos dados, não sendo possível tirar proveito das relações temporais no conjunto de dados.

Neste caso, uma abordagem possível consiste em extrair informações temporais, por meio de algum tipo de pré-processamento temporal dos dados ou modelagem temporal. Essa abordagem será tratada no Capítulo 3.

Alternativamente, a fim de melhorar a análise de dados temporais, é possível modificar o funcionamento da SVM para incorporar raciocínio temporal. Durante o desenvolvimento deste projeto, diversas abordagens para incorporar raciocínio temporal à SVM foram encontradas na literatura: SVM por Quadrados Mínimos Recorrentes (SUYKENS; VANDEWALLE, 2000), SVM Dependente de Perfil (CAMPS-VALLS et al., 2007), SVM com *kernel* sequencial (WAN; CARMICHAEL, 2005; RUDZICZ, 2009; JAAKKOLA; HAUSSLER, 1999; CAMPBELL, 2002; WATKINS, 1999), SVM com *kernel* recursivo (HERMANS;

SCHRAUWEN, 2012) e *Support Vector Echo-State Machines* (SHI; HAN, 2007). Neste trabalho, as abordagens SVM com *kernel* recursivo e *Support Vector Echo-State Machines* serão abordadas, visto que ambas podem ser aplicadas para o problema de classificação e, portanto, parecem adequadas para a resolução do problema de segmentação de gestos.

Assim, esta seção apresenta os dois modelos de SVM que incorporam raciocínio temporal, de interesse no contexto deste trabalho: SVM com *kernel* recursivo (Seção 2.2.1) e *Support Vector Echo-State Machines* (Seção 2.2.2).

2.2.1 SVM com *Kernel* Recursivo

A SVM pode ser adaptada para processar dados temporais por meio da modificação da forma de funcionamento do *kernel*. Uma destas formas é aplicar a recursão na execução do mapeamento, usando dados anteriores no cálculo do *kernel* do dado atual, ou seja, no mapeamento de cada dado, a informação contida na série histórica composta pelos dados anteriores deve ser considerada.

Considerando uma série temporal definida como uma coleção de observações feitas sequencialmente ao longo do tempo, na forma $\{\boldsymbol{s}(1), \boldsymbol{s}(2), \cdots, \boldsymbol{s}(n)\}$ para n observações, um *kernel* recursivo opera sobre duas séries temporais dadas por $\{\boldsymbol{s_i}(1), \boldsymbol{s_i}(2), \cdots, \boldsymbol{s_i}(n)\}$ e $\{\boldsymbol{s_j}(1), \boldsymbol{s_j}(2), \cdots, \boldsymbol{s_j}(n)\}$ (HERMANS; SCHRAUWEN, 2012), visando usar o histórico da série para a análise do dado atual. Um *kernel* recursivo κ no tempo t é dado por

$$\kappa_t(s_i, s_j) = \langle \Phi(s_i(t), \Phi(s_i(t-1), \Phi(\cdots))) \cdot \Phi(s_j(t), \Phi(s_j(t-1), \Phi(\cdots)))\rangle.$$

onde $\Phi(\cdot,\cdot)$ é um mapeamento não linear, $s_i(t)$ é a entrada atual da série s_i, e $\Phi(s(t-1), \Phi(\cdots))$ corresponde a um mapeamento não linear aplicado recursivamente a todos os elementos do histórico da série temporal.

Hermans e Schrauwen (2012) apresentam alguns exemplos de *kernel* recursivo. Tais exemplos consideram dois parâmetros: σ_h para ponderar o mapeamento sobre o histórico da série, e σ_i para ponderar a entrada atual. A estabilidade desses *kernels* depende da escolha desses parâmetros, conforme discutido em Hermans e Schrauwen (2012). Também deve ser estabelecido um valor para um parâmetro η que define o tamanho da recursão. A escolha deste valor é uma decisão a ser feita durante a definição do modelo. Em tese, o *kernel* recursivo poderia ter uma profundidade de recursão infinita, ou seja, considerar todo o passado da série para o cálculo do *kernel* de cada elemento da série. Como essa abordagem é computacionalmente inviável, é preciso definir uma profundidade de recursão

adequada, capaz de capturar as informações relevantes no histórico da série com o menor custo computacional possível. A lista de *kernels* apresentada pelo referido autor inclui:

- *Kernel* recursivo linear:

$$\kappa_t(s_i, s_j) = \sigma_i^2 \sum_{j=0}^{\eta} \langle \sigma_h^{2j} s_i(t-j) \cdot s_j(t-j) \rangle.$$

- *Kernel* recursivo polinomial:

$$\kappa_t(s_i, s_j) = \left[\langle \sigma_i^2 s_i(t) \cdot s_j(t) \rangle + \sigma_h^2 \kappa_{t-1}(s_i, s_j) \right]^q.$$

- *Kernel* recursivo gaussiano (também conhecido como RBF):

$$\kappa_t(s_i, s_j) = \exp\left(-\frac{\|s_i(t) - s_j(t)\|^2}{2\sigma_i^2} \right) \exp\left(\frac{\kappa_{t-1}(s_i, s_j) - 1}{\sigma_h^2} \right).$$

É importante notar que a SVM com *kernel* recursivo funciona da mesma forma que a SVM tradicional. As diferenças estão nas modelagens recursivas para as funções *kernel*, e na necessidade de que cada dado do conjunto de treinamento contenha pelo menos η elementos anteriores para que a recursão possa ser realizada. Ou seja, os resultados para os primeiros η elementos da série devem ser desprezados. Neste trabalho, η é definido por inspeção.

2.2.2 *Support Vector Echo-State Machines*

Outra estratégia para incorporar processamento temporal à SVM é chamada *Support Vector Echo-State Machine* (SVESM). De acordo com Shi e Han (2007), essa abordagem utiliza um reservatório (*reservoir*) – uma grande rede neural com neurônios conectados de forma esparsa e aleatória – para processar a informação temporal. O reservatório é capaz de processar informações temporais por considerar os estados anteriores no cálculo do estado atual, de acordo com:

$$\begin{aligned} x(i+1) &= \text{tansig}\left(\boldsymbol{W_x} \cdot \boldsymbol{x_i} + \boldsymbol{W_{in}} \cdot s(i) + \boldsymbol{v_{i+1}} \right) \\ y_i &= \langle \boldsymbol{w} \cdot x(i) \rangle + b \end{aligned}$$

onde *tansig* denota a função tangente hiperbólica; $s(i)$ é a entrada da SVESM, ou seja, a entrada do reservatório; $\boldsymbol{x_i}$ denota variáveis de estado no reservatório, que são usadas

como entrada para a SVM; y_i é a saída da SVM e, portanto, a saída do modelo como um todo; $\boldsymbol{W_{in}}$ representa o conjunto de pesos da camada de entrada do reservatório; $\boldsymbol{W_x}$ representa o conjunto de pesos internos do reservatório; $\boldsymbol{w}$ representa o conjunto de pesos da camada de saída do modelo, que, neste caso, corresponde aos pesos da SVM; e b é o bias da SVM. Neste caso, o reservatório é representado pelos estados x_i e pelos pesos de entrada e internos no reservatório, que são gerados de forma aleatória, ou seja, não são treinados.

O reservatório usado na SVESM deve respeitar uma condição chamada propriedade de estado de eco (*echo-state property*), que determina que o efeito de entradas e estados anteriores nos estados futuros deve desaparecer gradualmente ao longo do tempo. Tal propriedade pode ser satisfeita através da divisão da matriz de pesos internos pelo seu raio espectral – que corresponde ao maior autovalor absoluto da matriz (VOS, 2013).

Como na SVESM o reservatório pode substituir a função *kernel* da SVM tradicional, a entrada da SVM é processada pelo reservatório e submetida a uma SVM com *kernel* linear, ou seja, o peso na camada de saída $\boldsymbol{w}$ é definido pelo treinamento da SVM.

Para construir a SVESM, alguns parâmetros devem ser considerados: número de neurônios no reservatório; taxa de conectividade, que determina a quantidade de conexões no reservatório; taxa de vazamento, que determina por quanto tempo uma entrada tem influência nos estados do reservatório – quanto maior a taxa de vazamento, menor a influência de uma entrada nos estados futuros do reservatório; e uma janela descartada, que determina uma porção inicial do vídeo para a qual não são obtidas respostas de classificação, uma vez que se esperar incorporar informação sobre a história da série.

A SVESM foi inicialmente usada por Shi e Han (2007) para previsão de séries temporais. Neste trabalho, porém, pretende-se investigar seu uso para classificação de padrões com aspectos temporais. A Figura 5 ilustra o funcionamento da SVESM, focando no seu uso para classificação (ou seja, com o uso de um classificador, neste caso, implementado por uma SVM), e não para previsão conforme proposto por Shi e Han (2007).

2.3 Considerações Finais

Este capítulo apresentou uma descrição sobre SVM e algumas adaptações que incorporam raciocínio temporal à técnica, que na sua forma clássica considera os dados como independentes e identicamente distribuídos.

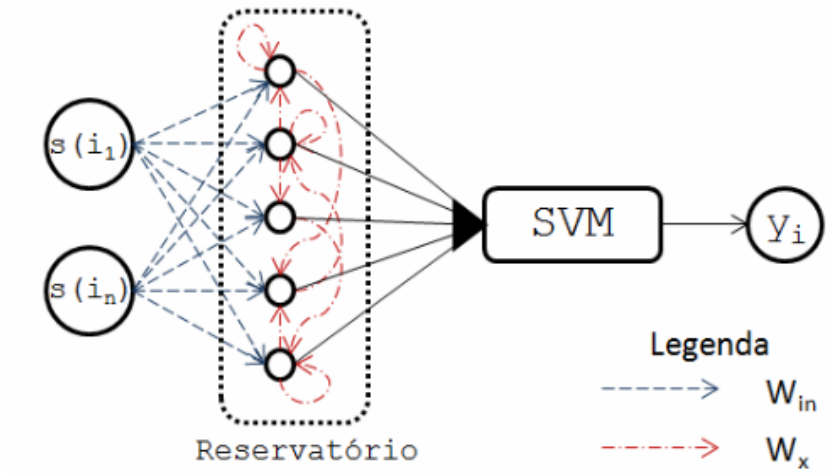

Figura 5 – *Funcionamento da SVESM: cada elemento i da série, com n atributos, é submetido a um reservatório e o estado do reservatório é submetido como entrada a uma SVM. O reservatório possui pesos na camada de entrada (W_{in}) e pesos internos (W_x).*

As técnicas apresentadas neste capítulo para incorporar raciocínio temporal às SVMs consistem na substituição do *kernel* utilizado na SVM tradicional por um *kernel* que considera aspectos temporais dos dados. No caso do *kernel* recursivo, dados do histórico da série são considerados para calcular o valor do *kernel* para cada dado da série. Já no caso da SVESM, um reservatório capaz de capturar aspectos temporais dos dados substitui o *kernel* tradicional.

Tais técnicas modificam a SVM, exigindo algum esforço em termos de compreensão e implementação. Outra opção para considerar aspectos temporais dos dados utilizando SVM é realizar um pré-processamento de forma a representar a dinâmica temporal dos dados para cada ponto da série analisada. Algumas formas para extrair características temporais dos dados serão apresentadas no Capítulo 3.

3 Representação Temporal

Conforme já mencionado neste texto, uma das formas de tratar eficientemente a informação temporal presente em um conjunto de dados é executar procedimentos de pré-processamento capazes de criar um espaço de características alternativo ao original, criando uma representação espacial do tempo na qual os aspectos temporais possam ser explorados mesmo a partir da aplicação de técnicas de análise que não incorporam raciocínio temporal. Um método bastante usado para este fim é a Reconstrução do Espaço de Fases, que permite representar um elemento de uma série temporal a partir da representação do estado do sistema no momento de geração daquele elemento.

Essa organização permite criar vetores a partir de cada elemento da série original, que podem ser diretamente usados por métodos convencionais de análise ou podem ser usados para extração de outras características a partir da submissão desses novos vetores a outros procedimentos.

Este capítulo apresenta a Reconstrução do Espaço de Fases (Seção 3.1) e alguns outros métodos para extração de características temporais (Seção 3.2), de frequência (Seção 3.3) e de tempo-frequência (Seção 3.4) que devem ser aplicados sobre séries temporais janeladas e, portanto, são aplicados sobre dados reconstruídos utilizando a Reconstrução do Espaço de Fases com $\tau = 1$, que, neste caso, corresponde a uma janela retangular simples.

3.1 Reconstrução do Espaço de Fases

O espaço de fases é um conceito baseado na teoria de sistemas dinâmicos que corresponde a um espaço de vetores no qual cada vetor especificado no espaço representa um estado do sistema, e vice versa (KANTZ; SCHREIBER, 2004). Assim, é possível estudar a dinâmica do sistema através do estudo da dinâmica no espaço de fases correspondente.

As séries temporais geralmente são compostas de escalares, de forma que cada escalar seja um dado da série. Para analisar a dinâmica da série, é preciso compor os vetores de

espaço a partir desses dados. Essa composição dos vetores é chamada de Reconstrução do Espaço de Fases (KANTZ; SCHREIBER, 2004).

Nesse contexto, a Reconstrução do Espaço de Fases se baseia na ideia de que uma série temporal é uma sequência de medidas escalares que depende do estado atual do sistema:

$$s(n) = S(\boldsymbol{x}(n\Delta t)) + \mu_n,$$

onde $s(n)$ é o n-ésimo ponto da série, composto por uma função S sobre o estado x no tempo $n\Delta t$ – com Δt sendo a unidade de deslocamento temporal para cada novo ponto da série – e uma medida de ruído μ_n. Assim, a reconstrução do espaço de fase em m dimensões é formada por vetores $\boldsymbol{s(n)}$ – um para cada elemento da série original – dada por:

$$\boldsymbol{s(n)} = (s(n-(m-1)\tau), s(n-(m-2)\tau), \cdots, s(n-\tau), s(n)), \tag{3.1}$$

onde τ é o atraso de tempo. De forma geral, pode-se dizer que o vetor reconstruído $\boldsymbol{s(n)}$ é equivalente ao estado $\boldsymbol{x}_n$ se a dimensão m for suficientemente grande (KANTZ; SCHREIBER, 2004). Observando a equação 3.1, é possível notar que uma reconstrução do espaço de fases com $\tau = 1$ corresponde à aplicação de uma janela retangular simples de tamanho m sobre o dado. A Figura 6 ilustra a Reconstrução do Espaço de Fases considerando $m = 3$ e $\tau = 2$.

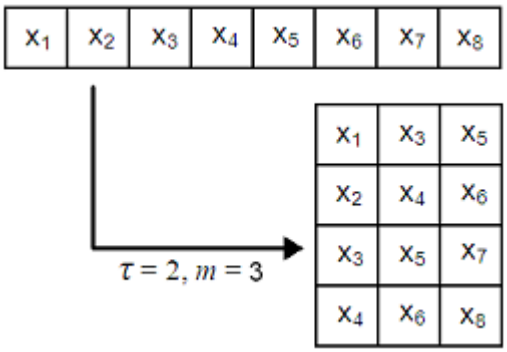

Figura 6 *– Reconstrução do Espaço de Fases de um vetor composto por oito elementos, considerando $m = 3$ e $\tau = 2$.*

Dado que se quer representar a informação temporal que rege o comportamento da série, é importante escolher adequadamente os parâmetros m e τ, visto que eles definem a informação histórica a ser usada na representação do comportamento de um sistema.

No caso da determinação da dimensão do espaço de fases, a partir de um determinado

valor de m, todo valor $m' > m$ fornece uma representação fiel do espaço de fases da série temporal. Porém, quanto maior o valor de m', maior a quantidade de informação redundante, podendo reduzir o desempenho de muitos algoritmos que objetivam analisar a série reconstruída, como algoritmos de previsão e classificação (KANTZ; SCHREIBER, 2004).

No que diz respeito à técnicas de escolha do parâmetro τ, um valor pequeno pode gerar um espaço que assume que quase não há dependência entre elementos sucessivos, podendo gerar informações redundantes; um valor muito grande, porém, pode gerar um espaço de reconstrução de fases que assume forte correlação entre elementos sucessivos, podendo causar perda de informação sobre a série (FRASER; SWINNEY, 1986; KANTZ; SCHREIBER, 2004).

Existem duas abordagens para escolher esses parâmetros: (a) utilizar técnicas de análise para estimar os melhores valores para determinada série (KANTZ; SCHREIBER, 2004); e (b) realizar a reconstrução variando tais parâmetros e avaliar o impacto de cada reconstrução no desempenho dos algoritmos que serão utilizados para analisar a série temporal. A segunda abordagem é a utilizada no presente trabalho.

3.2 Extração de Características no Domínio do Tempo

Phinyomark et al. (2009) usa algumas características no domínio do tempo para representar sinais de Eletromiografia (*Electromyography* – EMG). As características exploradas neste trabalho são calculadas a partir dos valores de amplitude do sinal, que, neste caso, corresponde à série temporal. As seguintes características foram consideradas:

- Soma dos valores absolutos (SVA):

$$SVA = \sum_{n=1}^{N} |s(n)|, \tag{3.2}$$

  onde N é o tamanho do sinal.

- Valor Médio Absoluto (*Mean Absolute Value* – MAV):

$$MAV = \frac{1}{N} \sum_{n=1}^{N} |s(n)|. \tag{3.3}$$

- Valor Médio Absoluto Ponderado por uma Função Janela (*Modified Mean Absolute Value 1* – MMAV1):

$$MMAV1 = \frac{1}{N}\sum_{n=1}^{N} w_n |s(n)|,$$

onde

$$w_n = \begin{cases} 1, & \text{se } 0.25N \leq n \leq 0.75N. \\ 0.5, & \text{caso contrário.} \end{cases}$$

- Raiz Quadrada Média (*Root Mean Square* – RMS):

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} |s(n)|^2}.$$

- Integral Quadrada Simples (*Simple Square Integral* – SSI):

$$SSI = \sum_{n=1}^{N} |s(n)|^2. \tag{3.4}$$

- Variância (Variance – VAR):

$$VAR = \frac{1}{N-1}\sum_{n=1}^{N} |s(n)|^2. \tag{3.5}$$

3.3 Características no Domínio da Frequência: Análise de Fourier

A análise de Fourier é um método clássico para análise espectral, ou seja, um método que decompõe um sinal em suas frequências constituintes. No caso da análise de Fourier, especificamente, realiza-se a decomposição de um sinal em uma soma de componentes sinusoidais (senos e cossenos) com frequências, amplitudes e fases apropriadas (BLOOMFIELD, 2000).

A análise de Fourier é baseada na transformada de Fourier. A transformada de Fourier contínua para uma função $f(t)$ relativa ao tempo t é dada por:

$$\hat{f}(\omega) = \int_{\infty}^{\infty} e^{-2\pi iwt} f(t),$$

onde ω representa a frequência em Hertz (KAISER, 2011). Neste caso, o valor $\hat{f}(\omega)$ representa a amplitude e a fase de um componente sinusoidal com frequência ω.

Para processamento digital, sinais contínuos precisam ser amostrados em intervalos discretos de tempo. Para realizar a análise de Fourier nesses sinais amostrados, é preciso usar a Transformada de Fourier Discreta (*Discrete Fourier Transform* – DFT). A DFT de um sinal $x(n)$, com $n = 0, 1, \cdots N-1$, é dada por:

$$X(m) = \sum_{n=0}^{N-1} x(n) e^{-i2\pi mn/N} \tag{3.6}$$

onde N é o número total de pontos no sinal e $m = 0, 1, \cdots, N-1$ (SEMMLOW, 2004). Cada valor de $X(m)$ contém um número complexo que representa a amplitude e a fase de um componente sinusoidal do sinal $x(n)$.

A análise de Fourier discreta é uma extensão do procedimento para análise contínua. Para sinais discretos, é preciso realizar uma amostragem do sinal, feita através do cálculo do espectro em diversos intervalos múltiplos de T, onde $T = 1/f_s$ e f_s é a frequência de amostragem do sinal. Teoricamente, a série de Fourier discreta de um sinal periódico e simétrico tem tamanho infinito e toda a informação relevante está contida no intervalo de frequência de 0 a $f_s/2$ (chamada de frequência de Nyquist) (SEMMLOW, 2004).

Além disso, sinais de tamanho infinito devem ser janelados para permitir sua análise computacional. O janelamento consiste em multiplicar o sinal por um formato de janela específico. A forma mais simples de janelamento utiliza a janela de formato retangular e consiste em considerar todos os pontos fora da janela como iguais a zero. Porém existem outras formas de janelamento, como a janela triangular, que suaviza os valores do sinal próximos à borda da janela (SEMMLOW, 2004). Neste trabalho, a análise de Fourier é aplicada a segmentos do sinal, janelados utilizando Reconstrução do Espaço de Fase com atraso igual a 1, o que equivale a uma janela retangular.

Conforme mostrado na equação 3.6, a saída da transformada de Fourier discreta é uma sequência de N elementos, no qual cada elemento é um número complexo que armazena informações sobre a amplitude e a fase do componente sinusoidal com a frequência correspondente. O módulo de cada número complexo representa a amplitude do componente sinusoidal, enquanto o argumento do número complexo representa o ângulo de

fase do mesmo componente. Neste trabalho, utiliza-se a amplitude dos componentes com frequências de 0 a $f_s/2$ para criar uma nova representação do sinal.

Para ilustrar a aplicação da transformada de Fourier discreta a um sinal contínuo, considere o sinal formado pela sobreposição de dois sinais senoidais dado por:

$$s_c(t) = sin(2 * \pi * f_1 * t) + sin(2 * \pi * f_2 * t) \tag{3.7}$$

onde f_1 é a frequência da primeira senoide, igual a 2Hz, e f_2 é a frequência da segunda senoide, igual a 4Hz. Para aplicar a transformada de Fourier discreta, é necessário realizar uma amostragem do sinal. Considerando uma frequência de amostragem de 20Hz, o sinal contínuo é amostrado como mostra a Figura 7.

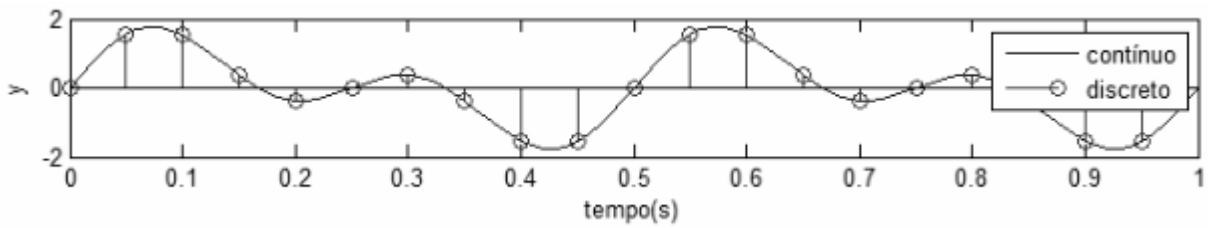

Figura 7 *– Sinal contínuo amostrado a uma frequência de 20Hz.*

Ao aplicar a transformada de Fourier, obtém-se uma representação dos componentes sinusoidais do sinal original. Essa representação consiste em um vetor de números complexos, a partir do qual é possível extrair a amplitude e a fase dos componentes. As amplitudes e fases relacionadas ao sinal da Figura 7 são apresentadas na Figura 8. Nota-se que a transformada de Fourier consegue extrair corretamente as amplitudes dos componentes sinusoidais originais – 2Hz e 4Hz – associados à fase correspondente – -90, visto que eram senoides.

A transformada de Fourier discreta n-dimensional também foi utilizada neste trabalho para obter uma nova representação do sinal. A transformada n-dimensional de um sinal $f(x, y)$, na sua forma contínua, é dada por:

$$f(w_1, \cdots, w_n) = \int_{\Re_n} e^{-2\pi i(w_1 t_1 + \cdots + w_n t_n)} f(t_1, \cdots, t_n) dt_1 \cdots dt_n, \tag{3.8}$$

onde w_i é a frequência espacial associada a cada direção t_i em $\Re_n$. Neste caso, a função $f(w_1, w_2 \cdots w_n)$ é representada por componentes sinusoidais com diferentes amplitudes e fases, um para cada frequência w_i, relativos a cada $f(t_i)$ (OSGOOD,).

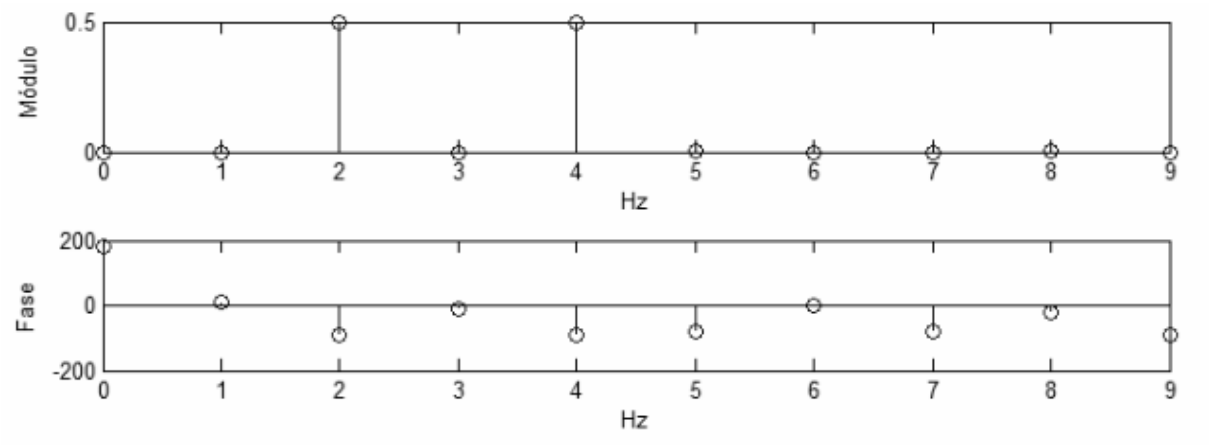

Figura 8 *– Amplitude e fase dos componentes sinusoidais do sinal original.*

3.4 Características no Domínio Tempo-Frequência: Análise Wavelet

A transformada de Fourier captura características do sinal no domínio da frequência, porém não é capaz de extrair características temporais dos dados. A transformada Wavelet é capaz de descrever um sinal nos domínios de tempo e frequência, sendo capaz de extrair características de um sinal cuja frequência se altera ao longo do tempo.

A transformada Wavelet contínua é dada por:

$$W(a,b) = \int_{-\infty}^{\infty} s(t) \frac{1}{\sqrt{|a|}} \psi * (\frac{t-b}{a})\, \mathrm{d}t, \tag{3.9}$$

onde $*$ indica a operação de conjugação complexa, b representa uma função sobre $s(t)$ e a varia a escala de tempo de uma função de sondagem ψ, uma função que é somada ou multiplicada visando modificar o sinal $s(t)$. No caso da transformada Wavelet, a função de sondagem é também chamada função Wavelet devido à sua forma oscilatória. O termo $\frac{1}{\sqrt{|a|}}$ é um fator de normalização que garante que a energia seja a mesma para todos os valores de a e b. Os coeficientes de Wavelet $W(a,b)$ descrevem a correlação entre o sinal original e o sinal com várias translações e escalas, ou seja, a similaridade entre o sinal e o mesmo sinal em determinada combinação de escala a e posição b (SEMMLOW, 2004).

Para sinais discretos, usa-se a transformada Wavelet discreta (*Discrete Wavelet Transform* – DWT), que restringe a variação de translação e escala em potências de 2. A transformada discreta pode ser implementada através do uso de dois filtros – um filtro passa-baixa $G(z)$ e um filtro passa-alta $H(z)$ – em um banco de filtros (SEMMLOW, 2004; SUBASI, 2007). O filtro passa-baixa corresponde à transformada Z de um filtro g e deve

satisfazer a condição dada por:

$$G(z)G(z^{-1}) + G(-z)G(-z^{-1}) = 1.$$

O filtro passa-alta correspondente é dado por:

$$H(z) = zG(-z^{-1}).$$

A transformada Z usada para gerar o filtro passa-baixa consiste em uma generalização da transformada de Fourier discreta, capaz de analisar sinais não periódicos (OPPENHEIM et al., 2009). A transformada Z de um sinal $x(n)$ é dada por:

$$X(m) = \sum_{-\infty}^{\infty} x(n)z^{-n}. \tag{3.10}$$

onde z é uma variável complexa. Ao comparar a equação 3.6 com a equação 3.10, nota-se que a transformada de Fourier é uma transformada Z com $|z| = 1$ (OPPENHEIM et al., 2009).

Cada filtro gera, respectivamente, coeficientes de aproximação (cA) e coeficientes de detalhe (cD), conforme ilustrado na Figura 9.

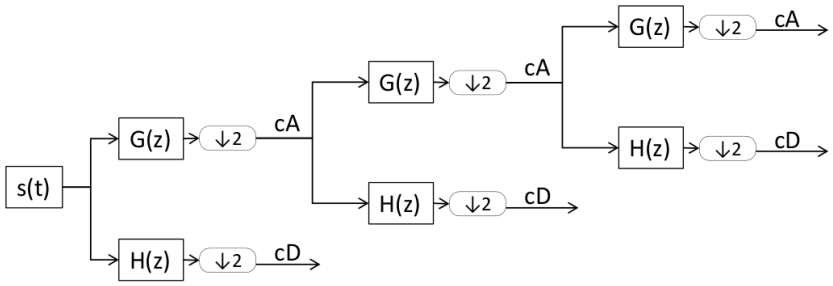

Figura 9 *– Representação da decomposição usando a transformada wavelet.*

Como existem características redundantes nos componentes gerados pela convolução do sinal com os filtros, é possível reduzir a amostragem desses componentes (*downsampling*), de forma que cada um tenha a metade do tamanho do sinal original (SEMMLOW, 2004).

Escolher os filtros adequados para um banco de filtros de Wavelet pode ser uma tarefa difícil, já que os filtros devem respeitar diversos pré-requisitos, como permitir a reconstrução precisa do sinal. Também pode ser desejável que os filtros deem origem a

coeficientes cA e cD ortogonais (SEMMLOW, 2004).

Uma família de filtros muito popular para realizar análise Wavelet que atende a tais pré-requisitos é a família Daubechies. Os filtros utilizados neste trabalho pertencem à família Daubechies com 4 coeficientes (SEMMLOW, 2004). Os coeficientes para o filtro passa-baixa $G(z)$ são dados por:

$$G(z) = \frac{[(1+\sqrt{3}), (3+\sqrt{3}), (3-\sqrt{3}), (1-\sqrt{3})]}{8}. \tag{3.11}$$

Os coeficientes do filtro passa-alta devem ser calculados de forma a garantir que cA e cD sejam ortogonais (SEMMLOW, 2004).

Em geral, a análise wavelet é feita utilizando vários níveis de decomposição do sinal. Inicialmente, o sinal original é decomposto gerando coeficientes de aproximação e coeficientes de detalhe, conforme ilustrado na Figura 9. Depois, os coeficientes de aproximação são decompostos da mesma forma, gerando outro nível de coeficientes de aproximação e de detalhe. Neste trabalho, os testes consideraram coeficientes de primeiro, segundo e terceiro nível.

Finalmente, segundo Lima e Coelho (2011) e Subasi (2007), para utilizar a análise wavelet como forma de obter uma nova representação do sinal, em geral faz-se uso de estatísticas calculadas sobre os vetores de coeficientes. Neste trabalho, aplica-se média e/ou desvio padrão dos coeficientes de aproximação do último nível de decomposição avaliado e dos coeficientes de detalhe de todos os níveis de decomposição avaliados para representar o sinal analisado via wavelet.

Para exemplificar a transformada Wavelet, considere um sinal contínuo formado pela sobreposição de dois sinais senoidais, dado por:

$$s(t) = sin(2 * \pi * f_1 * t) + sin(2 * \pi * f_2 * t) \tag{3.12}$$

onde f_1 é a frequência da primeira senoide, igual a 2Hz, e f_2 é a frequência da segunda senoide, igual a 45Hz.

Aplicando a transformada de Fourier nesse sinal, conforme apresentado na Seção 3.3, e considerando uma frequência de amostragem igual a 100Hz, obtém-se as frequências das senoides que constituem o sinal: 2Hz e 45Hz. Com o uso da transformada Wavelet no sinal, obtém-se os coeficientes de aproximação (cA) e os coeficientes de detalhe (cD). Reconstruindo os sinais usando apenas os coeficientes de aproximação e aplicando

a transformada de Fourier nesse sinal reconstruído, é possível perceber que os coeficientes de aproximação mantêm apenas as informações das frequências mais baixas do sinal – neste caso, a frequência de 2Hz. Da mesma forma, usando apenas os coeficientes de detalhe e aplicando a transformada de Fourier no sinal reconstruído a partir desses coeficientes, nota-se que os coeficientes de detalhe mantêm apenas as informações das frequências mais altas do sinal – neste caso, a frequência de 45Hz. Neste caso, como o sinal foi amostrado a 100Hz, a transformada Wavelet é capaz de obter informações de 0 a 50Hz, sendo que os coeficientes de aproximação mantêm as informações de 0 a 25Hz e os coeficientes de detalhe mantém as informações de 25Hz a 50Hz. Esse processo é apresentado na Figura 10.

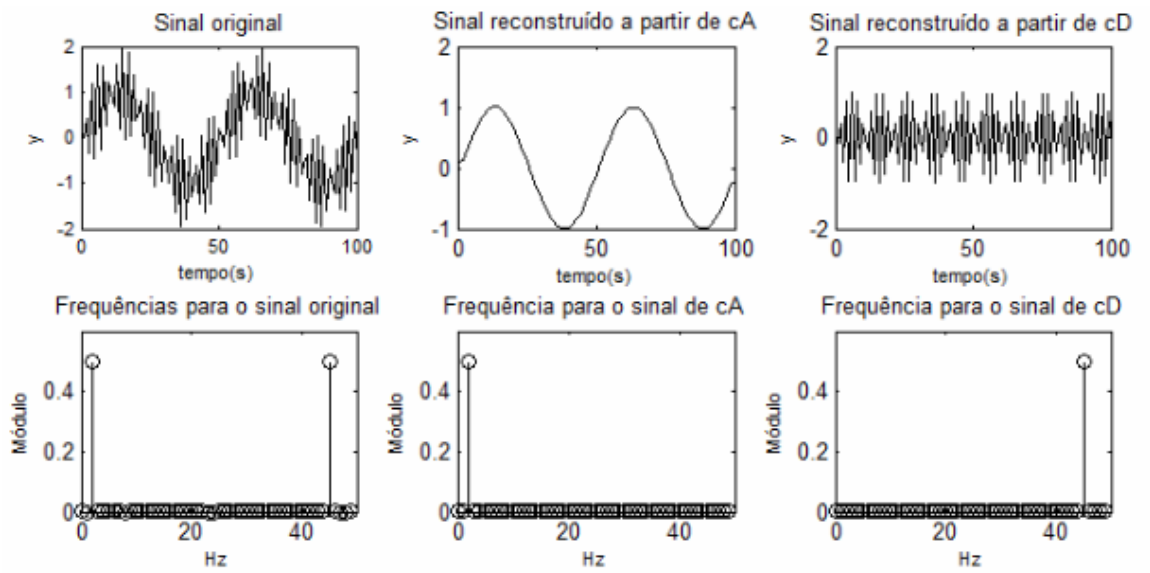

Figura 10 – *Sinal original, sinal reconstruído a partir dos coeficientes de aproximação e sinal reconstruído a partir dos coeficientes de detalhe (acima) e as respectivas frequências obtidas para cada sinal usando transformada de Fourier (abaixo).*

3.5 Considerações Finais

Este capítulo apresentou métodos para representação temporal dos dados, de forma a incorporar a dinâmica temporal do sistema representado por esses dados.

Todos os métodos apresentados são baseados na Reconstrução do Espaço de Fases, que consiste em representar um dado da série através de uma sequência não consecutiva de pontos, incluindo o dado a ser representado e alguns outros pontos da série, determinados pelos parâmetros de dimensão incorporada e de atraso.

A partir dessa reconstrução, que pode inclusive dar origem a um janelamento simples

dos dados caso o parâmetro de atraso seja igual a um, é possível extrair características nos domínios do tempo, frequência e tempo-frequência.

A representação temporal dos dados permite que as características temporais dos dados sejam consideradas, sem que haja a necessidade de modificar o método de análise utilizado (neste caso, SVM). Em geral, construir uma representação temporal é mais simples do que investigar um método que incorpore raciocínio temporal e ainda possibilita explorar diversos métodos de análise diferentes.

Por essas razões, a representação temporal dos dados é uma alternativa interessante e muito utilizada para a análise de problemas temporais.

4 Análise Automatizada de Gestos

Neste trabalho, aprendizado de máquina, implementado por meio de SVM e considerando características temporais, é aplicado à tarefa de segmentação das fases dos gestos. A análise das fases dos gestos é crucial para estudar a sincronia entre fala e gestos, permitindo o desenvolvimento de teorias sobre tal relação (GULLBERG, 2006). Atualmente, pesquisadores que realizam estudos dos gestos executam um trabalho manual de transcrição de gestos registrados em vídeos, a fim de produzir dados referentes à segmentação das fases dos gestos para suportar a análise da conversação ou do discurso. Segundo Quek et al. (2002), todo o processo de transcrição manual de um minuto vídeo – incluindo a transcrição e segmentação de diversos componentes, como fala, gestos, postura e expressões faciais – pode levar até 10 dias.

Durante o desenvolvimento deste trabalho, uma revisão sistemática[1] sobre análise automatizada de gestos, focando nos aspectos temporais da análise, foi realizada. Na organização do conteúdo da revisão sistemática foram criadas categorias para classificar os trabalhos analisados de acordo com a sua aplicação. As categorias consideradas foram: análise de um vocabulário pré-definido de gestos; análise de gesticulação; e análise psicolinguística.

Os estudos aplicados à análise de um vocabulário pré-definido de gestos são os mais comuns na literatura, devido à sua utilização para interação humano-computador em tarefas específicas (KIM et al., 2007; CHOI et al., 2008; CHEN et al., 2011; WONG; CIPOLLA, 2006; CORRADINI, 2002; YUAN, 2008; LI; GREENSPAN, 2007; LAMAR et al., 1999; WERAPAN; CHOTIKAKAMTHORN, 2004; NGUYEN; BUI, 2010). Neste capítulo, porém, são destacadas as linhas de trabalho focadas na análise de gestos naturais que acompanham a fala, usada na análise de gesticulação e na análise psicolinguística, visto que é nesse contexto que a tarefa de segmentação dos gestos é aplicada. Na Seção 4.1, uma discussão sobre trabalhos

[1] A revisão completa pode ser encontrada sob a forma de relatório técnico (MADEO; PERES, 2012).

focados na análise de gestos naturais é apresentada, dividida em: trabalhos visando a análise de gesticulação com foco no estudo da estrutura e da classificação automatizada dos gestos (Seção 4.1.1); e trabalhos aplicados à análise psicolinguística (Seção 4.1.2). Na Seção 4.2, a discussão se restringe apenas aos trabalhos focados especificamente na segmentação de fases dos gestos.

4.1 Estudos dos gestos e Análise Psicolinguística

Atualmente, pesquisadores de diversas áreas do conhecimento – como linguística, psicologia, psicolinguística, antropologia, sociologia, medicina – se dedicam aos chamados "estudos dos gestos". A área de estudos dos gestos é uma área interdisciplinar que visa estudar o uso das mãos e outras partes do corpo para fins comunicativos. Quando tais gestos com fins comunicativos são realizados junto à fala, podem ser chamados de gesticulação (MCNEILL, 1992).

Na literatura, foram encontrados trabalhos dedicados a análises automatizadas de diversos aspectos da gesticulação. Uma das principais áreas de aplicação da análise de gesticulação é a análise psicolinguística.

A psicolinguística visa entender o processo pelo qual as pessoas produzem e compreendem a comunicação (GARNHAM, 1994). Assim como diversos tipos de análises realizadas na área de estudos dos gestos, a análise psicolinguística pode considerar uma análise da gesticulação e usa transcrições como base para as análises. Caso os pesquisadores estejam interessados em analisar o uso de gestos na comunicação, a segmentação das fases dos gestos se torna parte importante da tarefa de transcrição.

Fora do contexto da análise psicolinguística, os trabalhos em análise da gesticulação focam a análise da estrutura e da classificação do gesto, sendo aplicados a diversas áreas de estudos dos gestos.

A teoria da área de estudos de gestos é necessária para entender o processo de produção dessas transcrições. Tais transcrições incluem, como elementos principais relativos ao gesto, a descrição visual dos gestos (a posição das mãos, o movimento, etc.), a segmentação dos gestos em fases e o tipo dos gestos.

A segmentação das fases dos gestos proposta por Kendon (1980) considera que existem unidades gestuais – períodos que iniciam quando o braço se afasta do corpo e termina quando ele retorna a uma posição de descanso. As unidades gestuais podem ser divididas

em fases:

- preparação: as mãos se movem para a posição na qual o movimento começa;
- *hold* pré-*stroke*: pequena pausa ao final da preparação, mantendo a posição e configuração das mãos;
- *stroke*: movimento que expressa o significado do gesto – é a única fase obrigatória no gesto, as demais são opcionais;
- *hold* pós-*stroke*: pequena pausa ao final do *stroke*, mantendo a posição e configuração das mãos;
- retração: retorno das mãos à posição de descanso.

Já sobre os tipos de gestos, McNeill (1992) define os possíveis tipos[2] como:

- icônico: associado a objetos e eventos concretos, cuja forma se assemelha a uma imagem relacionada ao conceito representado.
- metafórico: similar ao gesto icônico, mas representa ideias e conceitos abstratos.
- *beats*: marca o ritmo da fala. Não carrega significado semântico e sua função é revelar o que é importante ao discurso narrativo. Outro diferencial desse tipo de gesto é que, por representar apenas uma batida, pode-se considerar que ele tem apenas duas fases, como: cima/baixo, esquerda/direita, etc.
- coesivo: conecta partes do discurso separadas temporalmente, mas com temas relacionados. É usado para retomar um assunto após fazer um comentário, por exemplo.
- dêitico: gestos de apontar, usados para indicar objetos e eventos concretos ou localizar referências para abstrair conceitos no espaço gestual.

Os trabalhos que focam na análise de gesticulação fora do contexto da análise psicolinguística são apresentados na Seção 4.1.1. Já os trabalhos cujo escopo pertence à análise psicolinguística são abordados na Seção 4.1.2.

[2]McNeill (2005) cita os tipos como "dimensões"do gesto, de forma que um gesto pode conter uma ou mais dimensões.

4.1.1 Análise de gesticulação

A análise de gesticulação pode ser considerada mais complexa do que a análise de um vocabulário pré-definido porque, em geral, ao analisar um vocabulário pré-definido, é possível analisar semelhanças na forma do gesto – configuração das mãos, desenho do movimento, etc.; no caso de análise de gesticulação, pode ser necessário analisar gestos que compartilhem semelhanças em seus significados, mas que não tenham necessariamente o mesmo formato.

Esse problema pode ser exemplificado pelo trabalho de Kettebekov (2004) que visa reconhecer três tipos de gestos do tipo *beat*: *beat* simples, que apenas marca o ritmo da fala; *beat* icônico, quando o *beat* é, na verdade, parte de um gesto icônico ou representa, por si só, um objeto ou evento concreto; e *beat* transicional, quando o *beat* marca a mudança na estrutura do discurso – por exemplo, quando o falante interrompe o seu relato para fazer um comentário sobre o assunto e depois volta ao relato original.

No trabalho de Kettebekov (2004), mais de 30 minutos de palestras foram utilizados para extrair 1128 gestos do tipo *beat* – 476 para treinamento e 652 para teste. Os gestos para o treinamento foram segmentados manualmente com o auxílio da ferramenta *Gesture Analysis Tool* (KETTEBEKOV et al., 2002) e foram extraídas características de velocidade e aceleração das mãos e velocidade e aceleração das mãos em relação à cabeça. Modelos ocultos de Markov (*Hidden Markov Model* – HMM) foram utilizados para classificar os gestos, obtendo $73,5\%$ de acerto para *beat* simples, $56,3\%$ para *beat* icônico e $62,4\%$ para *beat* transicional. Os autores também propõem o uso de HMM considerando características da voz relacionadas aos picos de entonação vocal, para auxiliar na classificação dos gestos. Nesse caso, os autores obtiveram $82,9\%$ de acerto para *beat* simples, $74,8\%$ para *beat* icônico e $68,6\%$ para *beat* transicional. Os resultados mostram que a análise de características da voz pode auxiliar na análise da gesticulação, visto que existe uma forte relação entre gestos e fala.

Tal hipótese também é corroborada em Kettebekov et al. (2005) (versão estendida e aprimorada de Kettebekov et al. (2002)), que visa reconhecer gestos dêiticos com os significados "contornar" e "apontar", além de reconhecer as fases de preparação e retração associadas a esses gestos. Os dados foram captados durante transmissões de previsão do tempo, com um conjunto de 446 gestos, segmentados utilizando a ferramenta *Gesture Analysis Tool* e contendo 1262 segmentos (pertencentes às classes "contornar", "apontar", preparação e retração) para o treinamento e 1876 segmentos para o teste. Utilizando apenas características de velocidade e aceleração e HMM como em Kettebekov (2004), os

autores obtiveram uma média de $72,4\%$ de acerto. Ao utilizar características da voz, como em Kettebekov (2004), obtém-se uma média de $75,6\%$ de acerto foi obtida. Os autores ainda sugerem o uso de uma rede bayesiana para combinar os resultados dos modelos que analisam a voz com os modelos que analisam o gesto e, neste caso, obtiveram uma média de $84,2\%$ de acerto.

Também é possível trabalhar a ideia oposta: a análise da gesticulação pode auxiliar na análise da fala ou da voz. Tal hipótese é considerada em Eisenstein et al. (2008), que analisa gestos co-verbais para melhorar a precisão da análise de correferência, que visa relacionar pares de sintagmas nominais com mesmo significado. O conjunto de dados considera vídeos de pessoas descrevendo o funcionamento de dispositivos mecânicos, considerando 1137 sintagmas nominais. Para compor a representação dos gestos, diversas características baseada nas posições das mãos em cada gesto são extraídas. A correferência é decidida por meio de um modelo probabilístico chamado Fusão de Modalidade Condicional (*Conditional Modality Fusion* – CMF) que considera a representação dos gestos e características verbais. Utilizando apenas características verbais, o modelo obtém um desempenho de $0,7945$ em termos de área sob a curva caracterÃstica de operaÃ§Ã£o do receptor (*Receiver Operating Characteristic* – ROC). Ao utilizar CMF para considerar características dos gestos, obtém-se um desempenho de $0,8226$. Nota-se, portanto, que é possível utilizar a informação sobre os gestos para auxiliar em análises relativas a outros modos de comunicação, como fala ou tom de voz, por exemplo.

Outros trabalhos investigam a estrutura temporal dos gestos. É o caso dos trabalhos de Wilson e Bobick (2000), que utiliza a estrutura temporal do gesto para analisar gestos utilizados por maestros para conduzir orquestras; Swaminathan et al. (2008), que analisa o ritmo dos gestos de condução de orquestra visando identificar características da música conduzida.

Wilson e Bobick (2000) consideram a segmentação das fases de gestos de regência musical. No caso exemplificado nesse artigo, o gesto é composto das fases "baixo"e "cima", além da posição de descanso. A imagem é segmentada utilizando o algoritmo *Expectation-Maximization* (EM) e a sequencia de pixels que fazem parte da silhueta do sujeito na imagem compõem o vetor de representação da imagem. Para classificar uma imagem como pertecente a cada classe, usa-se uma HMM treinada por meio do algoritmo EM. Os resultados são apresentados por meio de gráficos e podem ser vistos no artigo original (WILSON; BOBICK, 2000). Outro ponto interessante do sistema proposto é que existe uma fase de treinamento no início de cada execução do sistema, visando adaptá-lo para cada

usuário.

Por fim, Swaminathan et al. (2008) visa identificar, a partir dos gestos, a forma de articulação da música interpretada – *legato* ou *staccato* – e o andamento (velocidade do compasso) da música, considerando vídeos de maestros iniciantes com marcadores nos ombros, cotovelos, pulsos e batutas para estimar a posição tridimensional de cada ponto e gerar características de velocidade e direção dos movimentos. Os autores propõem uma abordagem baseada em Rede Bayesiana Dinâmica (*Dynamic Bayesian Network* – DBM) para segmentar os *strokes* realizados e, posteriormente, inferir a articulação e o andamento da música. Os resultados são apresentados por meio de gráficos e podem ser vistos no artigo original (SWAMINATHAN et al., 2008).

4.1.2 Análise Psicolinguística

Algumas iniciativas encontradas durante a revisão sistemática propõem métodos automatizados para extração de características que possam ser utilizadas na análise psicolinguística. A ideia é gerar características que possam ser relacionadas a determinados aspectos do discurso, de forma a automatizar alguma tarefa da análise.

O trabalho de Bryll et al. (2001), por exemplo, visa identificar a presença de períodos de *hold* nos gestos. Nesse caso, os autores usam o termo *hold* para denotar pausas no movimento das mãos, sem diferenciar a fase de *hold* das posições de descanso dentro de um gesto. Esse estudo considera um vídeo de 4668 *frames*, no qual o primeiro participante deve discutir com o segundo participante um plano de ação para capturar uma família de vombates[3] inteligentes que invadiram um hotel abandonado. Os autores propõem uma heurística para detecção automática de *holds* baseada no cálculo de energia, a partir da mudança da posição tridimensional das mãos durante determinada janela de tempo, obtendo até $85,8\%$ de acerto.

Outros trabalhos apenas visam gerar gráficos de características extraídas automaticamente para facilitar a análise manual dos vídeos. É o caso de Chen et al. (2002), Quek et al. (2002), e Quek (2004), que apenas usam gráficos apresentando a posição das mãos ao longo do tempo, a quantidade de esforço realizado pela mão ao longo do tempo e a presença de *holds*, extraída conforme Bryll et al. (2001), para que os psicolinguistas possam relacionar tais características a reparos no discurso, estrutura do discurso e *catchment*[4],

[3] Vombates são marsupiais originários da Austrália que cavam sistemas de tocas e túneis.

[4] *Catchment* é um conceito que declara que temas de discurso coesos são relacionados por um conjunto de imagens recorrentes e, consequentemente, a características gestuais recorrentes.

respectivamente.

Xiong et al. (2002) propõem um método visando detectar movimentos simétricos de forma automatizada. A detecção de movimentos simétricos é baseada no cálculo da correlação entre os sinais obtidos a partir das posições das mãos direita e esquerda. Posteriormente, a existência de movimentos simétricos, extraídos automaticamente, é relacionada à estrutura do discurso.

Já Xiong et al. (2003) e Xiong e Quek (2006) apresentam métodos visando detectar movimentos oscilatórios ou cíclicos. Para realizar esse tipo de detecção, os autores usam a Transformada de Fourier Janelada (*Windowed Fourier Transform* – WFT) e a Transformada Contínua de Wavelet (*Continuous Wavelet Transform* – CWT). Xiong e Quek (2006) apresentam os resultados por meio de gráficos, indicando que a CWT produz melhores resultados na detecção de movimentos oscilatórios que a WFT. Depois, os autores relacionam a ocorrência de tais movimentos com as unidades do discurso.

Na área de análise psicolinguística, também são encontradas análises multimodais, ou seja, que consideram outros modos de comunicação, como conteúdo da fala e tom de voz. Eisenstein (2008) trata, dentre outros tópicos, da segmentação do discurso por assunto, considerando características dos gestos e do conteúdo da fala para determinar, por meio de uma estratégia bayesiana, a probabilidade de determinado trecho do vídeo ser considerado corretamente um tópico de discurso. Os resultados indicam que combinar gestos e conteúdo da fala pode fornecer melhores resultados do que analisar cada modo separadamente (EISENSTEIN, 2008). Chen et al. (2004), por sua vez, utilizam uma abordagem multimodal baseada em características gestuais, vocais e conteúdo da fala para identificar os limites de uma sentença. O modelo de linguagem é baseado em um modelo estatístico, enquanto os modelos gestual e de prosódia (tom de voz) usam árvores de decisão bayesianas. Os resultados dos modelos são integrados por meio de uma HMM. Para o melhor classificador utilizado isoladamente, o erro é de $53,2\%$. Para o classificador integrado, reduz-se o erro para $42,09\%$.

4.2 Segmentação das Fases dos Gestos

Na área de estudos dos gestos, existem várias iniciativas relacionadas à segmentação das fases dos gestos (KENDON, 2005; KITA et al., 1998; MCNEILL, 2005). Em relação à automação desta tarefa, porém, apesar da existência de várias iniciativas correlatas na literatura, foram encontrados apenas dois estudos que focam na segmentação de todas as

fases do gesto. De qualquer forma, é possível notar que existe um interesse recente na automação da tarefa, já que estudos encontrados são recentes.

Os dois estudos que focam a segmentação de todas as fases do gesto são Martell e Kroll (2007) e Ramakrishnan (2011). Ambos se concentram no reconhecimento das fases dentro de unidades gestuais, ou seja, realizam a classificação das fases já desconsiderando os períodos de descanso do vídeo. Ramakrishnan (2011) identifica as posições de descanso utilizando uma heurística baseada nas posições mais frequentes do vídeo, atingindo 87% de precisão e 4% de falsos positivos. Em Martell e Kroll (2007), o procedimento para segmentação das unidades gestuais não foi descrito.

O trabalho de Martell e Kroll (2007) considera como conjunto de dados um vídeo de 22 minutos de um professor lecionando. A partir das unidades gestuais já segmentadas, os autores aplicam HMM para segmentar as fases (preparação, *stroke*, *hold* e retração), obtendo, respectivamente, *F-scores* de 54%, 59%, 36% e 67%. Como os resultados para a detecção de *hold* não são satisfatórios, Martell e Kroll (2007) realizam um novo teste considerando apenas preparação, *stroke* e retração[5], obtendo *F-scores* de 56%, 68% e 79%.

Já Ramakrishnan (2011) usa o sistema de câmeras Vicon e luvas de dados, gerando 10 minutos de gesticulação de três usuários diferentes. A abordagem é baseada na detecção de pontos de inflexão (pontos de transição entre duas fases) por meio da análise de mudanças bruscas nos perfis de velocidade e aceleração. Após encontrar os pontos de mudança, cada ponto é classificado como início de uma fase de preparação, *stroke*, *hold* e retração, usando SVM. Como os resultados são baseados em segmentos corretos, e não em *frames*, a métrica para avaliação adotada pelo autor é diferente: se um segmento for classificado corretamente, mesmo que haja um desvio no início ou no fim do segmento, ele é considerado correto. Considerando essa métrica, Ramakrishnan (2011) obtém *F-scores* de 74.31%, 79.89%, 95.28% e 85.71% para preparação, *stroke*, *hold* e retração, respectivamente.

Além desses trabalhos, existem estudos que focam em tarefas específicas dentro da segmentação das fases dos gestos. Bryll et al. (2001) realizam a detecção de *holds* através de um método heurístico baseado na quantidade de esforço das mãos, conforme descrito na Seção 4.1.2, obtendo 82% de precisão e 86.4% de sensibilidade. Gebre et al. (2012), por sua vez, realizam a detecção de *strokes*, a partir de um vídeo de 3,6 minutos, com um total de 988 *frames* contendo *stroke* e 4453 *frames* que não contém *strokes*. Cada

[5]Neste caso, a fase de *hold* é incorporada às demais fases. Por exemplo, um pré-*stroke hold* pode ser considerado parte da fase de preparação, enquanto um pós-*stroke hold* pode ser considerado parte da fase de *stroke*.

frame é representado por um vetor contendo informações de posição e velocidade do *frame* analisado e de três *frames* anteriores e posteriores. Um classificador de regressão logística é utilizado, obtendo uma média de precisão de $47,24\%$ e sensibilidade de $34,41\%$, resultando em um *F-score* de $38,71\%$.

Por fim, Wilson et al. (1996) considera dois vídeos, com cerca de quatro minutos cada, nos quais os participantes devem contar uma experiência na qual eles estiveram em perigo. Para cada *frame*, dez coeficientes baseados na decomposição da imagem em autovetores são extraídos. Uma heurística, similar à usada por Ramakrishnan (2011), é utilizada para localizar os *frames* mais frequentes no vídeo, que são considerados candidatos a posições de descanso, e uma Máquina de Estados Finitos (*Finite State Machine* – FSM) com três estados – descanso (D), transição (T) e *stroke* (S) – é utilizada para determinar as fases de cada *frame*, considerando similaridade com uma posição de descanso, quantidade de movimento exibida e duração da fase. Caso um gesto contenha apenas a estrutura $\langle$ D-S-D $\rangle$, é considerado bifásico. Caso contenha a estrutura $\langle$ D-T-S-T-D $\rangle$, o gesto é considerado trifásico. Os resultados são apresentados por meio de representações gráficas e podem ser vistos no artigo original (WILSON et al., 1996). Nesse caso, a detecção das posições de descanso atinge aproximadamente 82% de precisão e 79% de sensibilidade[6].

4.3 Considerações Finais

Este capítulo apresentou uma breve revisão da literatura sobre análise automatizada de gestos, descrevendo algumas aplicações relacionadas a este trabalho, com uma seção específica sobre trabalhos correlatos que realizam a segmentação das fases dos gestos.

A revisão sobre aplicações correlatas destaca os trabalhos aplicados à análise da gesticulação e à análise psicolinguística. O presente trabalho é diretamente relacionado a essas duas aplicações, já que a segmentação das fases dos gestos analisa a gesticulação com o objetivo de fornecer informações que podem ser utilizadas na análise psicolinguística.

Já no caso de trabalhos correlatos, existem duas iniciativas principais com foco na tarefa de segmentação: Martell e Kroll (2007) e Ramakrishnan (2011). Apesar de não ser possível comparar diretamente a abordagem proposta nesta dissertação a esses trabalhos, visto que os conjuntos de dados utilizados são diferentes, eles podem ser usados como referência para avaliar os resultados obtidos pela abordagem proposta.

[6]Os valores de precisão e sensibilidade não são apresentados explicitamente em Wilson et al. (1996). Tais valores foram estimados pela autora desta dissertação a partir da análise gráfica da figura que compara o vídeo rotulado por uma pessoa e o vídeo rotulado automaticamente.

5 Experimentos, Resultados e Discussão

A partir do estudo realizado na área de SVM, considerando suas implementações tradicionais e implementações que incorporam características temporais, foi definido um conjunto de experimentos visando investigar o desempenho de tais técnicas no problema de segmentação das fases dos gestos.

Para esses experimentos, foram considerados tanto a análise de características temporais por meio de um pré-processamento de dados - usando Reconstrução do Espaço de Fases; quanto a incorporação de tais aspectos no próprio modelo SVM, usando SVM com *kernel* recursivo e SVESM. Para fins de estudo comparativo, também foram considerados os dados sem nenhum tipo de pré-processamento temporal dos dados.

O esquema gráfico apresentado na Figura 11 ilustra os procedimentos que compõem os experimentos descritos neste capítulo.

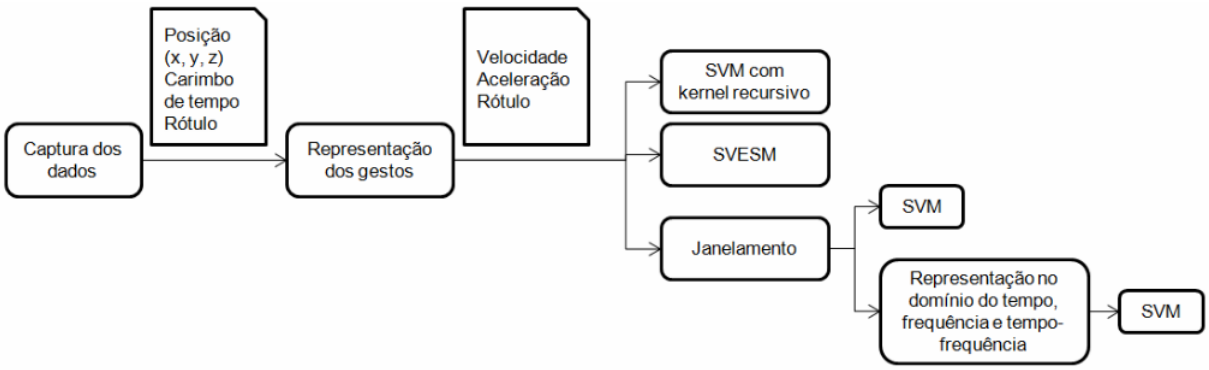

Figura 11 – *Esquema gráfico ilustrando os procedimentos que compõem os experimentos descritos neste capítulo para segmentação das unidades gestuais. Para a segmentação das fases dos gestos, segue-se o mesmo procedimento, porém considerando apenas SVM tradicional com dados janelados.*

Este capítulo contém a descrição dos conjuntos de dados utilizados nos experimentos, incluindo a forma de aquisição e extração dos dados (Seção 5.1), o método de realização dos experimentos (Seção 5.2) e os resultados obtidos (Seção 5.3).

5.1 Conjuntos de Dados

O objetivo deste trabalho é estudar a segmentação das fases dos gestos em situações de gesticulação natural, ou seja, situações nas quais os gestos são produzidos em conjuntos com a fala. Para obter dados referentes a esse contexto, foi escolhida a situação de "contação de histórias".

A contação de histórias foi solicitada a duas pessoas. Essas pessoas leram histórias em quadrinhos e, posteriormente, contaram a história presente nos quadrinhos em frente a um dispositivo de captura de dados, o Microsoft Kinect. A Tabela 1 apresenta algumas informações sobre os vídeos obtidos, incluindo a quantidade total de *frames* no vídeo (# Frames), a quantidade de *frames* de descanso (# Descanso) e a quantidade de *frames* correspondentes ao gesto (# Gesto), especificando também a quantidade de *frames* de cada fase – preparação (P), *stroke* (S), *hold* (H) e retração (R).

Tabela 1 *– Informações sobre os vídeos que compõem o conjunto de dados.*

Usuário	História	Duração	# *Frames*	# Descanso	# Gesto			
					P	S	H	R
A	1	60s	1747	698	1049			
					146	656	39	208
	2	40s	1264	493	771			
					180	431	54	106
	3	39s	1834	662	1172			
					279	535	150	208
B	1	37s	1112	286	826			
					236	262	193	135
	3	46s	1448	362	1086			
					338	389	144	215

Com o Microsoft Kinect, é possível obter: (a) as coordenadas tridimensionais (x, y, z) dos pontos de articulação escolhidos para rastrear os movimentos realizados por uma pessoa – as coordenadas são organizadas em um arquivo texto onde cada linha corresponde às coordenadas dos pontos obtidas em uma unidade de tempo de captação e um carimbo de tempo que mostra quando o dado foi capturado; (b) a imagem em formato RGB correspondente a um *frame* de vídeo, obtido em uma unidade de tempo de captação e um carimbo de tempo que mostra quando a imagem foi capturada. Assim, para cada *frame* de vídeo, são armazenadas as coordenadas tridimensionais de cada ponto de articulação de interesse e a imagem associada àquele *frame*[1].

[1]Devido a uma limitação do dispositivo, não é possível gravar simultaneamente o dado e a imagem, portanto, existe uma discrepância de milésimos ou até centésimos de segundo entre a imagem e o dado correspondente.

A partir das imagens dos *frames* e de seus carimbos de tempo, é possível atribuir manualmente a classe de cada dado, ou seja, a cada *frame*. Para executar esse processo de classificação, um analista examina as imagens em sequência, visando identificar a classe de cada *frame*. Para cada imagem examinada, o analista atribui uma classe na linha de dados correspondente ao carimbo de tempo da imagem no arquivo texto que contém as coordenadas de cada ponto de articulação.

É importante lembrar que o processo de segmentação das fases dos gestos é um processo subjetivo. Portanto, para considerar essa subjetividade na análise dos resultados obtidos no experimento, cada vídeo foi rotulado por dois analistas diferentes. Assim, é possível comparar o erro obtido pelo aprendizado de máquina com a taxa de discordância apresentada quando dois analistas diferentes rotulam o mesmo vídeo. A porcentagem de discordância para as rotulações recebidas por cada vídeo do conjunto de dados é apresentada na Tabela 2.

Tabela 2 – *Discordância apresentada por dois analistas ao rotular cada um dos vídeos do conjunto de dados (em %).*

Usuário	História	Discordância para Unidades Gestuais	Discordância para Fases
A	1	4,01	10,07
	2	4,19	12,97
	3	3,82	14,61
B	1	4,06	17,12
	3	2,18	17,17

Os motivos para discordância entre dois analistas são diversos, porém os mais comuns incluem: discordância sobre o momento exato de início/fim de cada fase; períodos curtos de descanso, que podem ser interpretados como *hold* por um dos analistas; e períodos de *stroke* múltiplo, nos quais o *stroke* é composto por um movimento complexo, que pode ser visto como uma única fase de *stroke* ou como um período composto de diversas fases de preparação, *stroke* e *hold*.

Neste trabalho, os seguintes pontos de articulação são utilizados para compor a representação dos gestos:

- Mãos: os pontos de articulação correspondentes às mãos esquerda e direita fornecem a localização espacial das mãos durante a execução do gesto.

- Pulsos: a posição dos pulsos pode fornecer informação sobre movimentos internos de cada mão através da relação de sua posição com a posição do ponto de articulação da mão correspondente.
- Cabeça: a posição da cabeça pode ser usada como um ponto de referencial para análise da velocidade e distância das mãos com relação à cabeça.
- Tronco: o ponto de articulação correspondente ao tronco pode ser utilizado como referencial alternativo ao ponto de articulação referente à cabeça, já que essa pode executar gestos não vinculados à posição das mãos, enquanto o tronco, normalmente, se mantém estável em relação ao gesto.

Para representar os gestos, neste trabalho estão sendo considerado os movimentos realizados durante a contação de histórias, representados pela velocidade e aceleração assumida por cada ponto de articulação. Essas informações podem ser influenciadas pelo deslocamento corporal realizado pela pessoa que executa o gesto e esse deslocamento não deve fazer parte da representação do gesto em si. Assim, as posições da cabeça e do tronco são consideradas apenas para realizar uma normalização das posições das mãos e dos pulsos, por meio de:

$$\boldsymbol{p}_{norm} = \frac{\boldsymbol{p} - \boldsymbol{p}_t}{\|\boldsymbol{p}_c - \boldsymbol{p}_t\|},$$

onde $\boldsymbol{p}$ é a posição de um ponto de articulação, $\boldsymbol{p}_{norm}$ é a posição normalizada do ponto de articulação $\boldsymbol{p}$, $\boldsymbol{p}_t$ é a posição do tronco e $\boldsymbol{p}_c$ é a posição da cabeça[2].

A partir dos pontos normalizados, as informações de velocidade e aceleração foram calculadas para representar o movimento das mãos e dos pulsos. Primeiro calcula-se a velocidade vetorial dada por:

$$\boldsymbol{v}_{vet} = \frac{\boldsymbol{p}_i - \boldsymbol{p}_{i-d}}{t_i - t_{i-d}}, \tag{5.1}$$

onde $\boldsymbol{p}_i$ é o vetor que representa a posição de um ponto de articulação considerando os eixos x, y e z no i-ésimo *frame* do vídeo, t_i corresponde ao tempo (em centésimos de segundo) no qual ocorre o i-ésimo *frame* do vídeo, e d é o deslocamento em *frames* usado para calcular a velocidade. Segundo Ramakrishnan (2011), é possível reduzir o

[2]A normalização é feita para tornar a representação invariante à posição do usuário na imagem e também invariante à proximidade do usuário da câmera.

ruído gerado pelo cálculo da velocidade utilizando $d > 1$. Portanto, neste trabalho, foram considerados diferentes valores para d, visando reduzir o ruído nos dados.

A partir da velocidade vetorial, obtém-se a aceleração vetorial, que consiste em:

$$\boldsymbol{a}_{vet} = \frac{\boldsymbol{v}_i - \boldsymbol{v}_{i-1}}{t_i - t_{i-1}}.$$

Medidas escalares, obtidas a partir das medidas vetoriais, também foram utilizadas. Neste caso, a medida escalar, tanto para velocidade quanto para aceleração, é dada por:

$$\nu_{esc} = \sum_{i=1}^{N} \nu_i^2,$$

onde ν é a medida vetorial considerada, ou seja, velocidade ou aceleração.

Cada uma dessas medidas pode ser interpretada como um sinal: a sequência de medidas de velocidade escalar da mão direita para cada *frame* forma o sinal correspondente à velocidade escalar da mão direita dessa sequência de *frames*. A Figura 12 e a Figura 13 apresentam, respectivamente, os sinais da velocidade escalar da mão direita e da aceleração escalar da mão direita considerando um trecho do vídeo A1.

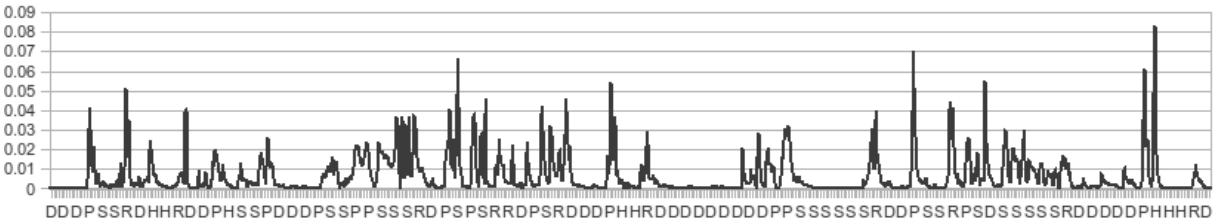

Figura 12 *– Sinal correspondente à velocidade escalar da mão direita.*

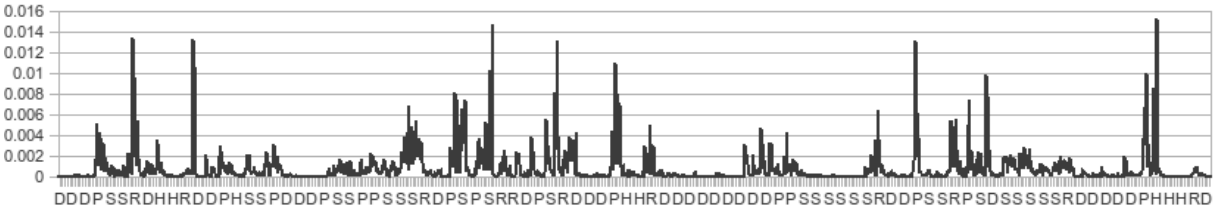

Figura 13 *– Sinal correspondente à aceleração escalar da mão direita.*

Assim, o conjunto de dados a ser analisado pela SVM é composto, para cada *frame*, por um vetor contendo a velocidade e/ou a aceleração vetorial ou escalar para mãos e/ou pulsos. Se considerarmos apenas medidas escalares, por exemplo, o vetor u que representa o dado consiste em:

$$u = [v_{me}, v_{md}, v_{pe}, v_{pd}, a_{me}, a_{md}, a_{pe}, a_{pd}], \tag{5.2}$$

onde v_{me} é a velocidade escalar da mão esquerda, v_{md} é a velocidade escalar da mão direita, v_{pe} é a velocidade escalar do pulso esquerdo, v_{pd} é a velocidade escalar do pulso direito, a_{me} é a aceleração escalar da mão esquerda, a_{md} é a aceleração escalar da mão direita, a_{pe} é a aceleração escalar do pulso esquerdo e a_{pd} é a aceleração escalar do pulso direito.

5.2 Descrição dos Experimentos

Todos os experimentos foram realizados usando a ferramenta Matlab com a *toolbox* Spider. Foi fixado o parâmetro $C = 100$, escolhido a partir de testes preliminares, considerando valores de C iguais a 1, 10, 100 e 1000, com melhor resultado obtido usando $C = 100$. Foram elaborados grupos de experimentos para atender o objetivo de analisar a eficiência dos modelos de SVM, a adequabilidade da representação escolhida para os dados e também da extração de características realizada sobre essa representação.

Esses grupos foram divididos em lotes de experimentos:

- Experimentos usando um algoritmo de SVM tradicional: o *kernel* utilizado foi o RBF, já implementado pela ferramenta Spider, e o parâmetro de abertura da RBF variou, inicialmente[3], de 2^{-2} a 2^4, com expoente variando de 1 em 1. Esse formato de experimento foi utilizado para avaliar o desempenho da SVM tradicional considerando: dados não janelados; dados janelados (ou seja, usando Reconstrução do Espaço de Fases com $\tau = 1$); dados usando Reconstrução do Espaço de Fases com $\tau = 2, \cdots, 8$; e usando uma representação de dados alternativa construída a partir da extração de características no domínio do tempo, frequência e tempo-frequência a partir dos dados janelados. Nos experimentos que consideram janelas, inicialmente foram consideradas janelas de até 80 *frames*[4].

- Experimentos considerando um algoritmo de SVM, fornecido pela Spider, com um *kernel* RBF recursivo implementado durante o desenvolvimento deste projeto. Neste

[3]Quando os melhores resultados foram obtidos com o menor ou a maior parâmetro considerado, novos testes foram realizados para aumentar o intervalo de parâmetros investigado, de forma que alguns experimentos consideraram intervalos maiores.

[4]Quando os melhores resultados foram obtidos com janelas de tamanho próximo a 80, janelas maiores foram consideradas.

caso, foram considerados os dados não janelados e os parâmetros investigados foram: parâmetro de abertura da RBF para o dado atual (de 2^{-2} a 2^4 com expoente variando de 1 em 1); parâmetro de abertura da RBF para o histórico da série (de 2^{-2} a 2^4 com expoente variando de 1 em 1); e tamanho da recursão (1, 5, 10, 20, 30, 40 e 50).

- Experimentos considerando uma implementação de SVESM[5]. Neste caso, foram considerados os dados não janelados e os parâmetros investigados foram: tamanho do reservatório (20, 40, 60, 80, 100, 200, 300, 400 e 500); taxa de vazamento (de 0,2 a 1 – em intervalos de 0,2); taxa de conectividade (de 0,2 a 1 – em intervalos de 0,2); e janela descartada (de 20 a 100 – em intervalos de 20).

- Experimentos visando avaliar a melhor estratégia multiclasse para realizar a segmentação das fases dos gestos. Foram consideradas as estratégias um contra um, um contra todos e hierárquica, usando SVM tradicional. Para as estratégias um contra um e um contra todos, o parâmetro de abertura da RBF variou de 2^{-2} a 2^4, com expoente variando de 1 em 1, e janelas de até 90 frames, conforme indicado por resultados nos experimentos anteriores. Para a estratégia hierárquica, o intervalo do parâmetro de abertura da RBF foi modificado para 2^2 a 2^5, com expoente variando de 1 em 1, de acordo com os resultados dos experimentos precedentes.

Para cada grupo de experimentos, vários modelos foram treinados e testados para atender às combinações dos diferentes parâmetros e características dos dados. A quantidade de modelos gerados para cada grupo ficará mais clara na Seção 5.3.

A estratégia de teste usada nos experimentos foi a estratégia *holdout*, que, neste caso, consiste em realizar o treinamento com dados de um vídeo e realizar a validação com dados de outro vídeo. Para os experimentos realizados neste capítulo, foram considerados os vídeos A1 para treinamento e A2 para validação.

5.3 Resultados

Os primeiros cinco experimentos – de #1 a #5 – visam explorar todas as variações da SVM para considerar os aspectos temporais dos dados no problema de classificar períodos de descanso de unidades gestuais.

[5]A implementação de SVESM foi adaptada para classificação a partir do código disponibilizado no site "Reservoir Computing" (http://reservoir-computing.org/software).

Os demais experimentos – #6, #7 e #8 – consideram o melhor método dentre os utilizados nos primeiros cinco experimentos para avaliar diferentes estratégias de classificação multiclasse, visando distinguir os *frames* de descanso, *hold*, *stroke*, preparação e retração dentre todos os *frames* do vídeo.

5.3.1 Experimento #1 – SVM tradicional com dados janelados

O primeiro experimento tem o objetivo de avaliar o desempenho da SVM tradicional para o problema de classificar *frames* entre pertencentes a um trecho de descanso ou a um trecho dentro de uma unidade gestual, considerando dados janelados. Por meio desse experimento, também é possível avaliar o desempenho utilizando dados não janelados, que correspondem às janelas compostas por apenas um *frame*.

Nesse experimento, diversas variáveis devem ser consideradas:

- Balanceamento dos conjuntos: os conjuntos de treinamento e validação podem ser bastante desbalanceados, dependendo da classe considerada. Nesse caso, é possível que balancear o conjunto de treinamento melhore o desempenho do classificador. Assim, as duas opções de treinamento são consideradas: com dados balanceados ou não balanceados.

- Deslocamento utilizado para extração de características: o valor da variável d, utilizada para determinar o deslocamento considerado para o cálculo da velocidade, segundo a equação 5.1. Os valores testados foram 1, 3 e 5.

- Representação vetorial ou escalar: a velocidade e a aceleração podem ser consideradas na sua forma vetorial ou escalar. O experimento inclui o uso dos dois tipos de variáveis para avaliar a melhor forma de representação.

- Informação a ser usada na representação: podem ser utilizadas as informações de velocidade, aceleração ou ambas para representar cada *frame*.

- Pontos de articulação: pode-se considerar informações das mãos, dos pulsos ou de ambos.

- Posição do rótulo: no caso de dados janelados, o rótulo pode corresponder ao último *frame* da janela, de forma que dados futuros não sejam considerados para a classificação do dado atual, ou ao *frame* no meio da janela, considerando dados passados e futuros. Como este trabalho não pretende realizar uma segmentação em tempo real, as duas abordagens são analisadas.

Ao considerar todas as 216 combinações possíveis dessas variáveis, a variação do número de *frames* por janela (1 a 80) e dos valores do parâmetro do *kernel* (de 2^{-2} a 2^4 com expoente variando de 1 em 1), seria necessário gerar 120.960 modelos de SVM. Devido ao grande número de testes e aos recursos computacionais necessários para executá-los, optou-se por dividir o experimento nas etapas descritas na Tabela 3.

Tabela 3 *– Descrição das etapas do experimento # 1.*

Etapa	Variáveis fixadas	Variáveis avaliadas	Variáveis selecionadas	Conjuntos de testes
# 1	Posição do rótulo: meio	Deslocamento, representação vetorial ou escalar, informação a ser usada na representação, pontos de articulação e tipo de conjunto	Deslocamento, representação vetorial ou escalar e tipo de conjunto	48
# 2	Posição do rótulo: meio; Deslocamento, representação vetorial ou escalar e tipo de conjunto selecionadas na etapa # 1	Informação a ser usada na representação e pontos de articulação	Informação a ser usada na representação e pontos de articulação	9
# 3	Deslocamento, representação vetorial ou escalar, informação a ser usada na representação, pontos de articulação e tipo de conjunto selecionadas nas etapas # 1 e # 2	Posição do rótulo	Posição do rótulo	2

Nessa estratégia, apenas 59 combinações de variáveis são analisadas, gerando 33.040 modelos de SVM – $27,3\%$ dos modelos necessários ao considerar todas as combinações possíveis de variáveis. Os resultados para cada etapa são apresentados nas Seções 5.3.1.1, 5.3.1.2 e 5.3.1.3.

5.3.1.1 Etapa #1

A primeira etapa do experimento #1 visa avaliar qual é o deslocamento ideal para extrair a característica de velocidade, se as características devem ser escalares ou vetoriais e se o conjunto de treinamento deve ser balanceado ou não. Para avaliar a forma de

extração e representação das características, é preciso considerar as duas características possíveis para os dois pontos de articulação considerados. Para minimizar o número de conjuntos de testes executados, porém, não são considerados nessa fase testes que usam ambas as características e/ou ambos os pontos de articulação.

Os testes são realizados considerando um conjunto de treinamento não balanceado e um conjunto de treinamento balanceado. Os melhores resultados, em termos de *F-score*, são apresentados na Tabela 4. É possível notar que não há diferença significativa nos resultados usando ou não balanceamento no conjunto de treinamento. Por esse motivo, optou-se por considerar os conjuntos originais, não balanceados. Essa opção facilita a comparação direta dos resultados com outras abordagens que não permitem o balanceamento de conjuntos, como SVM com *kernel* recursivo e a SVESM, que exigem que os dados sejam processados em sequência.

Tabela 4 – *Etapa #1 do experimento #1 com conjunto de treinamento não balanceado e balanceado, considerando deslocamento, tipo de característica, características e pontos de interesse – F-score dos modelos gerados.*

Conjunto de treinamento não balanceado								
	Mãos				Pulsos			
	Velocidade		Aceleração		Velocidade		Aceleração	
Deslocamento	Vet.	Esc.	Vet.	Esc.	Vet.	Esc.	Vet.	Esc.
1	0,828	0,811	0,733	0,731	0,774	0,787	0,684	0,710
3	0,843	**0,861**	0,798	0,767	0,823	0,851	0,781	0,768
5	0,836	0,859	0,802	0,755	0,815	0,818	0,801	0,783
Conjunto de treinamento balanceado								
	Mãos				Pulsos			
	Velocidade		Aceleração		Velocidade		Aceleração	
Deslocamento	Vet.	Esc.	Vet.	Esc.	Vet.	Esc.	Vet.	Esc.
1	0,827	0,806	0,731	0,739	0,779	0,796	0,688	0,707
3	0,843	**0,860**	0,790	0,768	0,824	0,850	0,781	0,787
5	0,838	0,859	0,805	0,764	0,816	0,825	0,802	0,813

Também é possível notar que os testes que consideraram deslocamento igual a 3 apresentam melhores resultados de forma geral, com resultados claramente superiores aos resultados obtidos com deslocamento igual a 1 e levemente superiores aos resultados obtidos com deslocamento igual a 5, em 10 dos 16 dos casos. Portanto, o deslocamento igual a 3 foi selecionado para os próximos testes.

Por fim, avalia-se o uso de características vetoriais ou escalares. Apesar de um desempenho superior das características vetoriais em alguns casos específicos, especialmente relacionados à característica de aceleração, foram obtidos resultados superiores com características escalares em $62,5\%$ dos casos. Adicionalmente, o melhor resultado obtido nesta

etapa do experimento ocorreu com o uso de características escalares – $86,1\%$ considerando velocidade das mãos com deslocamento igual a 3. Portanto, o tipo de característica escalar foi selecionado.

5.3.1.2 Etapa #2

Na segunda etapa do experimento, objetiva-se analisar qual combinação de características e pontos de interesse fornece melhores resultados. Os testes dessa etapa analisam o uso de ambas as informações e de ambos os pontos de articulação. Os melhores resultados são apresentados na Tabela 5 e indicam que os melhores resultados são obtidos ao utilizar as informações de velocidade e aceleração de mãos e pulsos em conjunto.

Tabela 5 *– Etapa #2 do experimento #1, considerando diferentes características e pontos de articulação – F-score dos modelos gerados.*

	Velocidade	Aceleração	Ambas
Mãos	0,861	0,767	0,854
Pulsos	0,851	0,768	0,870
Ambos	0,875	0,796	**0,877**

5.3.1.3 Etapa #3

Nas etapas anteriores, a posição central da janela foi escolhida como *frame* de interesse, ou seja, o rótulo para determinada janela corresponde ao rótulo do *frame* que ocupa a posição central da janela. A escolha foi baseada na hipótese de, para realizar a segmentação correta das fases, é interessante considerar *frames* passados e futuros em relação ao *frame* de interesse. A terceira etapa do experimento tem por objetivo testar essa hipótese, apresentando o desempenho do modelo quando o rótulo corresponde ao último *frame* da janela.

Com as mesmas variáveis utilizadas para obter o melhor resultado apresentado na Etapa #2, mudando apenas a posição do rótulo, o melhor *F-score* obtido é de $0,746$, $0,131$ inferior ao resultado do modelo que aplica o rótulo ao *frame* central da janela. Assim, o rótulo na posição central da janela foi escolhido para representar os dados.

Portanto, ao fim dessa etapa, os valores para algumas variáveis foram estabelecidos para serem usados nos demais experimentos:

- Tipo de conjunto: não balanceado;
- Deslocamento: 3;

- Tipo de característica: escalar;
- Característica: velocidade e aceleração;
- Pontos de articulação: mãos e pulsos;
- Posição do rótulo: meio.

O modelo com melhor desempenho, gerado com 81 *frames* na janela e parâmetro da RBF igual a 16, apresentou um *F-score* de $0,8774$, erro de classificação de $9,83\%$, precisão de $85,22\%$ e revocação de $90,41\%$.

Também é possível analisar o tipo de erro dentro de cada segmento e avaliar erros por segmento. Para realizar essa análise, considera-se que existem diferentes tipos de erro: erros de transição entre duas fases; e erros internos aos segmentos de determinada fase[6]. A partir disso, considera-se um segmento errado se: (a) mais de 20% do segmento for composto de erros internos; (b) mais de 40% do segmento for composto de erros de qualquer tipo.

Utilizando essa forma de análise, nota-se que, dos 116 *frames* classificados incorretamente, 46 *frames* são erros de transição, 58 *frames* são erros internos e 12 *frames* correspondem a segmentos inteiros classificados incorretamente. Dos 72 *frames* de gesto classificados como descanso, 22 são erros internos relacionados a *frames* que correspondem à fase de *hold*, 29 são erros internos relacionados à fase de *stroke* e 21 são erros de transição. Dos 44 *frames* de descanso classificados como gesto, 25 são erros de transição, 7 *frames* são erros internos e 12 *frames* correspondem a dois segmentos de descanso pequenos, com apenas quatro e oito *frames*.

Ao considerar erros por segmentos, de um total de oito segmentos de descanso e nove segmentos de gesto, os dois segmentos pequenos de descanso foram classificados incorretamente como gesto e um segmento de gesto foi classificado como descanso. Esse segmento de gesto classificado como descanso compreende 19 dos 22 *frames* de erro interno relacionados à fase de *hold*, 14 dos 29 *frames* de erro interno relacionados à fase de *stroke* e 9 dos 21 erros de transição relacionados a gestos classificados como descanso.

[6]Erros de transição são os erros que ocorrem durante a transição entre duas fases, quando os primeiros *frames* de uma fase são classificados como pertencentes à classe anterior ou quando os últimos *frames* de uma fase são classificados como pertencentes à classe posterior. No trabalho de Ramakrishnan (2011), tais erros são chamados de desvios. Já os erros internos são os demais erros que ocorrem dentro do segmento, e não na transição entre fases.

5.3.2 Experimento #2 – SVM tradicional com características derivadas a partir dos dados janelados

A partir dos dados janelados, é possível extrair características nos domínios de tempo, frequência e tempo-frequência. Neste trabalho, foram consideradas as características apresentadas no Capítulo 3: diversas características no domínio do tempo, análise de Fourier (domínio da frequência) e Wavelet (domínio da tempo-frequência).

5.3.2.1 Características no Domínio do Tempo

As características no domínio do tempo consideradas foram: soma dos valores absolutos; valor médio dos valores absolutos; valor médio absoluto ponderado por uma função janela; raiz quadrada média; integral quadrada simples; e variância. A Tabela 6 apresenta os resultados dos experimentos considerando tais características.

Tabela 6 *– F-score dos melhores modelos para cada característica no domínio do tempo.*

Característica	Erro	Janela	RBF
SVA	0,797	18	64
MAV	0,797	18	64
MMAV1	**0,799**	38	64
RMS	0,797	20	64
SSI	0,783	23	0,25
VAR	0,783	23	0,25

A partir dos resultados na Tabela 6, nota-se que nenhuma dessas características foi capaz de gerar um modelo com desempenho superior ao desempenho obtido usando dados janelados. Também é possível notar que algumas características geram o mesmo resultado para todos os modelos. É o caso da soma dos valores absolutos e do valor médio dos valores absolutos, que geram valores proporcionais: o valor médio corresponde ao valor da soma dividido por N, o número de *frames* na janela, conforme a equação 3.2 e a equação 3.3. O mesmo ocorre com a integral quadrada simples e a variância, que também geram valores proporcionais: o valor da variância corresponde ao valor da integral quadrada simples dividido por $N - 1$, conforme a equação 3.4 e a equação 3.5.

5.3.2.2 Características no Domínio da Frequência – Fourier

Neste trabalho, a análise de Fourier foi considerada de duas formas:

- Considerando o sinal como se fosse unidimensional: todas as dimensões do sinal são dispostas em sequência e processadas pela análise de Fourier simples.

- Considerando a análise n-dimensional: a análise de Fourier n-dimensional é usada para considerar as dimensões do sinal separadamente.

A Tabela 7 apresenta os resultados para cada forma. Apesar de ainda não apresentar resultados superiores aos obtidos no experimento #1, a análise de Fourier n-dimensional apresenta os melhores resultados dentre os modelos que consideram características derivadas dos dados janelados. Os resultados indicam, portanto, que é mais adequado tratar as dimensões de forma separada.

Tabela 7 *– F-score dos melhores modelos considerando análise de Fourier simples e n-dimensional.*

Fourier	Erro	Janela	RBF
Simples	0,787	26	4
n-dimensional	**0,819**	18	16

5.3.2.3 Características no Domínio Tempo-Frequência – Wavelet

Neste experimento, extraem-se medidas estatísticas sobre os coeficientes de aproximação e de detalhe gerados pela análise Wavelet usando Daubechies 4 para representar os dados. As medidas consideradas são média e desvio padrão. Tais medidas são consideradas isoladamente ou em conjunto, formando um vetor com média e desvio padrão de um conjunto de coeficientes. Também são considerados diferentes níveis de decomposição do sinal, conforme levantado no Seção 3.4. Para cada nível de decomposição, considera-se medidas estatísticas para cada conjunto de coeficientes de detalhe de todos os níveis e medidas estatísticas para os coeficientes de aproximação do último nível.

A Tabela 8 apresenta os resultados considerando diferentes níveis de decomposição e diferentes estatísticas. Os modelos gerados considerando características obtidas a partir da análise Wavelet não obtiveram desempenho superior ao modelo usando apenas os dados janelados. Nota-se que os melhores resultados foram obtidos com nível de decomposição 1, usando média e desvio padrão como estatísticas para representar o dado.

Tabela 8 *– F-score dos melhores modelos considerando análise Wavelet.*

Nível de Decomposição	Média	Desvio Padrão	Média e Desvio Padrão
1	0,787	0,765	**0,793**
2	0,777	0,764	0,776
3	0,764	0,776	0,776

5.3.3 Experimento #3 – Reconstrução do Espaço de Fases

A Reconstrução do Espaço de Fases permite a representação da série considerando diferentes atrasos. Neste trabalho, optou-se por realizar a reconstrução variando o parâmetro de atraso de 2 a 10 e a dimensão incorporada de 2 a 50. Caso os melhores resultados fossem obtidos com os maiores valores para os parâmetros, a busca seria ampliada.

A Tabela 9 apresenta os melhores resultados para cada valor de atraso, com dimensão incorporada e parâmetro da RBF correspondentes. Nota-se que nenhum modelo supera o modelo com $\tau = 1$ obtido pelo experimento #1. Também é possível notar que melhores resultados são obtidos com atrasos menores, não justificando uma extensão da busca considerando atrasos maiores que 10.

Tabela 9 *– Desempenho dos melhores modelos gerados com Reconstrução do Espaço de Fases.*

Atraso	Dimensão Incorporada	Parâmetro RBF	*F-score*
2	35	8	**0,8590**
3	28	8	0,8465
4	17	8	0,8473
5	10	4	0,8289
6	16	8	0,8214
7	7	4	0,8113
8	7	4	0,8016
9	7	4	0,7988
10	5	32	0,7940

A partir da Tabela 9, também é possível notar que os melhores resultados analisam janelas com pelo menos 41 e no máximo 82 *frames*, visto que no caso da Reconstrução do Espaço de Fases o tamanho da janela analisada é dado, aproximadamente, pelo tamanho da dimensão incorporada multiplicado pelo atraso. O melhor resultado foi obtido com $\tau = 1$ (experimento #1) com 81 *frames* na janela. O melhor resultado com $\tau = 2$ considerou $m = 35$, considerando 69 *frames* da série para representar cada dado. Para $\tau = 3$ e $\tau = 4$, respectivamente, considerou-se $m = 28$ e $m = 17$, analisando 82 *frames* e 65 *frames*.

5.3.4 Experimento #4 – *Kernel* Recursivo

Neste experimento, o *kernel* RBF recursivo foi utilizado a fim de incorporar a análise de aspectos temporais ao *kernel* da SVM. Esse *kernel* depende de três parâmetros: o parâmetro do *kernel* aplicado ao dado atual; o parâmetro do *kernel* aplicado recursiva-

mente ao histórico da série; e o tamanho da recursão da série. Como descrito na Seção 5.2, os parâmetros do *kernel* variaram de 2^{-2} a 2^{4} com expoente variando de 1 em 1 e foram testados tamanhos de recursão iguais a 1, 5, 10, 20, 30, 40 e 50.

Considerando todas as combinações possíveis desses parâmetros, foram executados 343 testes. A Tabela 10 contém um resumo dos dez melhores modelos obtidos, de acordo com o *F-score*. Os melhores resultados foram alcançados com valores baixos para os parâmetros do *kernel* (entre 0,25 e 1), com o parâmetro que pondera ao histórico da série maior do que o parâmetro que pondera o elemento atual da série. Também nota-se que os melhores resultados correspondem a tamanhos de recursão entre 5 e 20.

Tabela 10 *– Desempenho dos melhores modelos gerados com kernel recursivo: erro e F-score.*

Tamanho da Recursão	Parâmetro – atual	Parâmetro – histórico	Erro (%)	*F-score*
20	0,25	1	17,74	**0,8011**
10	0,5	1	17,84	0,8007
10	1	1	18,16	0,7993
20	0,5	1	17,98	0,7993
5	1	1	18,33	0,7975
10	0,25	1	18,40	0,7954
5	0,5	1	18,49	0,7947
20	0,25	0,5	17,50	0,7931
10	2	1	19,84	0,7877
5	2	0,5	19,20	0,7865

A Figura 14 mostra os melhores resultados para cada tamanho de recursão. Nota-se que os menores erros correspondem aos modelos gerados com tamanho de recursão igual a 10 e 20, indicando que melhores resultados poderiam ser encontrados nesse intervalo.

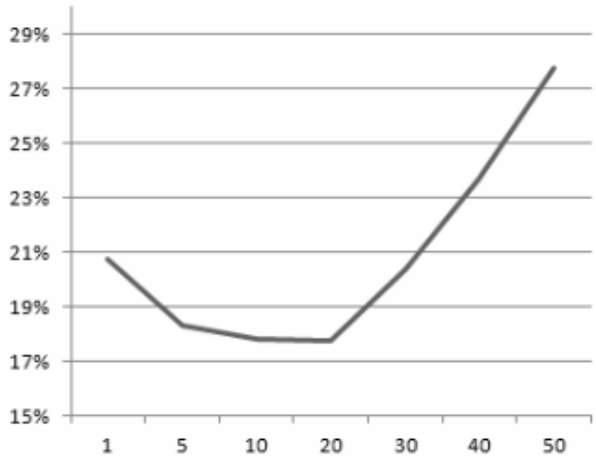

Figura 14 *– Erros dos melhores modelos para cada tamanho de recursão.*

Assim, novos testes foram realizados com tamanho de recursão variando de 11 a 19,

com os dois parâmetros do *kernel* variando de 2^{-2} a 2^{0}. A Tabela 11 apresenta os melhores resultados desse teste. Os resultados encontrados não foram significativamente melhores do que os encontrados na Tabela 10, e, portanto, não superam o resultado obtido com SVM tradicional e dados janelados.

Tabela 11 *– Desempenho dos melhores modelos gerados com kernel recursivo: erro e F-score.*

Tamanho da Recursão	Parâmetro – atual	Parâmetro – histórico	Erro (%)	*F-score*
19	0,25	0,5	15,95	**0,8062**
17	0,25	1	17,54	0,8040
17	0,25	0,5	15,29	0,8037
18	0,25	1	17,55	0,8036
14	0,25	1	17,58	0,8036
13	0,25	1	17,64	0,8032
15	0,25	1	17,75	0,8025
18	0,25	0,5	15,94	0,8024
16	0,25	1	17,77	0,8021
13	0,5	1	17,80	0,8018

5.3.5 Experimento #5 – *Support Vector Echo-State Machine*

No caso da SVESM, um reservatório é usado para realizar um mapeamento não linear dos dados, capaz de considerar informações de dados anteriores. Tal reservatório pode substituir o *kernel*, de forma que a SVM que processa os dados obtidos por meio do reservatório pode ser utilizada com *kernel* linear, como é o caso deste experimento.

Para criar o reservatório, alguns parâmetros devem ser considerados: o tamanho do reservatório; a conectividade do reservatório; a taxa de vazamento da memória; e a quantidade de dados que deve ser descartada no início do processamento usando o reservatório. Como descrito na Seção 5.2, tais parâmetros foram variados, respectivamente, nos valores 20, 40, 60, 80, 100, 200, 300, 400 e 500; de 0,2 a 1 – variando 0,2; de 0,2 a 1 – variando 0,2; e de 20 a 100 – variando 20. Considerando todas as combinações desses parâmetros, 720 modelos foram gerados.

O desempenho da SVESM é, de forma geral, pior do que o desempenho das demais técnicas estudadas. Os melhores resultados são obtidos com reservatórios menores. Dentre os 50 melhores resultados, 36% foram construídos descartando 80 *frames* de cada vídeo e 36% descartando 60 *frames*. É interessante notar que o número de *frames* descartados é similar ao número de *frames* obtido nos melhores modelos usando Reconstrução do Espaço de Fases. Nenhuma relação entre taxa de vazamento e taxa de conectividade com

melhores resultados foi encontrada.

O modelo com melhor desempenho foi obtido considerando um reservatório com 60 neurônios, 75% de conectividade, taxa de vazamento de 75% e descartando os primeiros 60 *frames* de cada vídeo. Tal modelo obteve um *F-score* de $0,7527$, com erro de classificação de 23% no conjunto de validação, $63,73\%$ de precisão e $91,90\%$ de revocação – desempenho inferior ao obtido pelo melhor modelo encontrado usando SVM tradicional com dados janelados.

5.3.6 Experimento #6 – Estratégia um contra um

A partir dos resultados obtidos nos experimentos anteriores, optou-se por aplicar SVM tradicional com dados janelados ao problema de segmentação das fases do gesto como um todo, incluindo as fases de preparação, *stroke*, *hold* e retração. Para isso, foi usada a estratégia um contra um, na qual são construídos classificadores binários para cada par de classes do problema multiclasse. Assim, para um problema com c classes, $c(c-1)/2$ classificadores são construídos. Neste caso, portanto, 10 classificadores são gerados. A resposta final da estratégia é decidida por votação.

Foram gerados diversos conjuntos de classificadores, fixando $C = 100$ e variando o parâmetro de abertura da RBF de 2^{-2} a 2^4 com expoente variando de 1 em 1, considerando janelas contendo entre 1 e 80 *frames*. As demais variáveis foram escolhidas conforme o melhor classificador obtido no experimento #1. Em cada conjunto de classificadores, todos os modelos usaram os mesmos parâmetros de abertura da RBF e tamanho das janelas. O melhor conjunto foi obtido considerando uma janela de 39 *frames* e parâmetro de abertura da RBF igual a 16. Conforme apresentado na Tabela 12, tal conjunto apresenta um *F-score* médio de $0,5255$, precisão de $59,39\%$, revocação de $52,78\%$ e $33,8\%$ de erro.

Tabela 12 *– Desempenho do melhor conjunto de classificadores usando a estratégia um contra um – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	70,72	87,87	0,7837
Preparação	62,80	51,50	0,5659
Stroke	74,93	58,24	0,6554
Hold	62,50	18,52	0,2857
Retração	26,02	47,76	0,3368
Média	59,39	52,78	0,5255

As posições de descanso compõem a classe para a qual o conjunto apresenta melhor desempenho, porém o *F-score* obtido neste caso é significativamente inferior ao *F-score*

obtido na classificação de posição de descanso no vídeo. O conjunto apresenta pior desempenho para as demais classes, o que é esperado devido à complexidade desses problemas. A classe *stroke* é a que apresenta melhor desempenho após posição de descanso. Tal resultado era esperado, visto que o *stroke* é entendido como o pico de movimento do gesto e apresenta um número considerável de exemplos. A preparação é a classe para a qual o conjunto obtém o terceiro melhor desempenho. Os piores desempenhos são obtidos para as classes *hold* e retração.

A Tabela 13 apresenta a matriz de confusão para o melhor conjunto de classificadores. Nessa tabela, é possível notar que $59,3\%$ dos *frames* de *hold* foram classificados como descanso – erro esperado devido à similaridade das duas classes. Os demais erros são referentes às classes de preparação, *stroke* e retração, que de fato compartilham características similares.

Tabela 13 *– Desempenho do melhor conjunto de classificadores usando a estratégia um contra um – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**413**	8	15	1	33
	P	49	**103**	39	0	9
Classe real	S	82	49	**251**	0	49
	H	32	0	12	**10**	0
	R	8	4	18	5	**32**

5.3.7 Experimento #7 – Estratégia um contra todos

Ao analisar o experimento #6, foi possível perceber uma redução da precisão na tarefa de identificação de posições de descanso ao usar a estratégia um contra um, com classificadores binários. Como os melhores resultados para essa tarefa, obtidos no experimento #1, consideraram a classificação de *frames* da classe descanso e da classe gesto, com gesto abrangendo todas as demais classes, neste experimento optou-se por testar a estratégia um contra todos.

A estratégia um contra todos gera, para c classes, c classificadores binários de forma que cada classificador C_i visa distinguir a classe c_i de todas as classes restantes. A resposta final da estratégia para um novo dado é a classe c_i correspondente à SVM que obtiver a maior saída.

Novamente, foram gerados diversos conjuntos de classificadores, fixando $C = 100$ e variando o parâmetro de abertura da RBF de 2^{-2} a 2^4 com expoente variando de 1

em 1, considerando janelas contendo entre 1 e 80 *frames*. As demais variáveis foram escolhidas conforme o melhor classificador obtido no experimento #1. Em cada conjunto de classificadores, todos os classificadores usaram os mesmos parâmetros de abertura da RBF e tamanho das janelas.

O melhor conjunto foi obtido considerando uma janela de 69 *frames* e parâmetro de abertura da RBF igual a 16. Conforme apresentado na Tabela 14, tal conjunto apresenta um *F-score* médio de $0,5277$, precisão de $61,47\%$, revocação de $51,13\%$ e $31,5\%$ de erro.

Tabela 14 *– Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	72,74	93,03	0,8164
Preparação	75,00	42,86	0,5455
Stroke	72,77	66,36	0,6942
Hold	61,54	19,05	0,2909
Retração	25,29	34,38	0,2914
Média	61,47	51,13	0,5277

Apesar de não obter desempenho médio significativamente superior à estratégia um contra um, a estratégia um contra todos obtém, conforme esperado, desempenho superior para a classe descanso. Também é observado desempenho superior para a classe *stroke*. Assim, o desempenho melhora para as classes que são identificadas com maior facilidade e que apresentam maior número de exemplos, de forma a diminuir o erro geral da abordagem.

A partir da matriz de confusão apresentada na Tabela 15, é possível notar que a natureza dos erros permanece similar, porém fica claro o aumento no desempenho para as classes de descanso e *stroke* e a redução para as classes preparação, *hold* e retração.

Tabela 15 *– Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**427**	6	25	0	1
	P	49	**84**	55	1	22
Classe real	S	78	22	**286**	4	41
	H	30	0	3	**8**	1
	R	18	0	24	0	**22**

5.3.8 Experimento #8 – Estratégia hierárquica

Outra estratégia para realizar a classificação multiclasse consiste em construir uma hierarquia de classificadores binários conforme descrito no Capítulo 1. Neste caso, a cada classificador, os dados classificados como positivos pelo classificador anterior são excluídos do próximo classificador. Isso implica que os erros dos primeiros classificadores não podem ser corrigidos pelos demais classificadores, de forma que um bom desempenho dos classificadores iniciais é crucial para o desempenho da estratégia.

O desempenho de cada um dos classificadores isolados é apresentado na Tabela 16. Neste caso, é possível perceber que os classificadores apresentam bons resultados. Ao utilizar a estratégia hierárquica, porém, os erros de precisão de um classificador prejudicam os índices de revocação das próximas classes a serem avaliadas. Assim, de forma geral, a precisão do classificador analisado isoladamente corresponde à precisão da classe na abordagem como um todo[7], mas a revocação fica pior quanto mais distante do início da hierarquia está o classificador.

Tabela 16 *– Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	85,22	90,41	0,8774
Preparação	70,54	90,10	0,7913
Stroke	72,10	85,93	0,7841
Hold	52,00	65,00	0,5778

A Tabela 17 mostra o desempenho da estratégia como um todo usando as métricas de precisão, revocação e *F-score*. Em relação às demais estratégias para classificação multiclasse, existe uma melhora no desempenho para as classes descanso e *stroke* e uma redução no erro, para $28,3\%$.

Para a classe descanso, os resultados indicam que a classificação binária entre descanso e gesto permite melhores resultados do que a combinação de resultados usada nas estratégias um contra um e um contra todos.

Para a classe *stroke*, uma hipótese pode ser formulada a partir da análise da matriz

[7]Na prática, devido à criação de janelas, esse resultado pode ser levemente diferente. Os resultados para os classificadores isolados consideram apenas janelas válidas processadas pelo algoritmo, enquanto os resultados para a abordagem consideram respostas para todos os *frames*, mesmo que não seja possível criar janelas para processar tais frames. Por exemplo, ao criar uma janela com 20 *frames*, é necessário ter 9 *frames* anteriores ao *frame* de interesse. Portanto, não é possível criar janelas correspondentes aos primeiros 9 *frames* do vídeo. Nesse caso, a resposta da classificação para um *frame* não avaliado é a classe correspondente ao primeiro classificador cujo tamanho de janela não possibilitou a criação de um dado janelado para o *frame*.

Tabela 17 – *Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	78,48	91,00	0,8428
Preparação	70,54	45,50	0,5532
Stroke	72,10	77,96	0,7492
Hold	52,00	24,07	0,3291
Retração	24,66	20,93	0,2264
Média	59,56	51,89	0.5401

de confusão, apresentada na Tabela 18, e das matrizes para as estratégias anteriores. O melhor desempenho do classificador de descanso faz com que menos *frames* de *stroke* sejam considerados como descanso, permitindo que o classificador de *stroke* classifique corretamente esses *frames*. Como não existe votação ou escolha da maior saída da SVM, quando um *frame* de *stroke* não é classificado como descanso, a possibilidade de que ele seja classificado corretamente pelo classificador especializado nessa classe é maior.

Para a classe de preparação, fica claro o impacto da estratégia hierárquica na revocação. Como 29 *frames* de preparação já foram classificados incorretamente como *descanso*, *hold* e *stroke* – como pode ser visto na Tabela 18 –, o classificador de preparação não dispõe de todos os *frames* de preparação para realizar a classificação corretamente. Já a classe de retração é dada pelos *frames* que não são classificados como preparação no último classificador. Como o objetivo do classificador é encontrar a classe de preparação, a retração absorve os erros de todos os demais classificadores, prejudicando sua precisão.

Tabela 18 – *Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**445**	8	21	7	8
	P	24	**91**	75	0	10
Classe real	S	35	18	**336**	5	37
	H	34	3	4	**13**	0
	R	29	9	30	0	**18**

Nessa estratégia hierárquica, foi possível testar diferentes tamanhos de janela para cada classificador. Foi possível notar que existe relação entre o tamanho médio das classes nos vídeos e o tamanho de janela usado para os classificadores com melhores resultados: o melhor classificador para períodos de descanso, em geral maiores nesse vídeo, usou uma janela de 81 *frames*; *hold* 17; o classificador de *stroke* 20; e o classificador de preparação e retração usa uma janela de 23 *frames*; com parâmetro da RBF respectivamente igual a

16, 8, 4 e 8.

Também é possível analisar os erros por segmento em relação às fases dos gestos. Neste caso, por razões de simplicidade, inicialmente analisa-se apenas se o maior número de *frames* dentro do segmento é classificado corretamente. Dessa forma, obtém-se a matriz de confusão apresentada na Tabela 19. No total, 18 segmentos foram classificados incorretamente de um total de 49. Todos os segmentos classificados incorretamente têm tamanho inferior a um segundo (30 *frames*) e, por esse motivo, mais propícios a serem prejudicados por erros de transição.

Tabela 19 *– Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**7**	1	0	1	0
	P	1	**8**	5	0	0
Classe real	S	2	0	**11**	0	0
	H	2	0	1	**3**	0
	R	2	1	1	2	**2**

5.4 Considerações Finais

Neste capítulo, foram apresentados os conjuntos de dados considerados neste trabalho, o método para obtenção de características e os experimentos realizados.

Os experimentos apresentados neste capítulo visam avaliar a aplicação de diferentes métodos que consideram representação ou raciocínio temporal. Os métodos de representação temporal dos dados considerados consistem em representar os dados por meio do uso de janelas de dados ou extrair características de tempo, de frequência ou de tempo-frequência a partir das janelas de dados e processar esses dados usando uma SVM tradicional. Os métodos que incorporam raciocínio temporal foram SVM com *kernel* recursivo e a SVESM, que substitui a função *kernel* por um reservatório.

Os primeiros experimentos deste capítulo – de #1 a #5 – tiveram por objetivo estudar o potencial de cada método para resolver o problema de classificação de um dado como pertencente a um período de descanso ou gesto. A partir desses experimentos, foi possível perceber que os métodos de representação temporal dos dados produziram resultados melhores que os métodos de raciocínio temporal.

Para a segmentação da unidade gestual, o melhor modelo obteve *F-score* de $0,8774$,

erro de $9,83\%$, precisão de $85,22\%$ e revocação de $90,41\%$. Retomando os trabalhos correlatos apresentados na Seção 4.2, nota-se que Ramakrishnan (2011) apresenta, para a mesma tarefa, um *F-score* de 0.913, com precisão de 87% e revocação de 96% e Wilson et al. (1996) obtém um *F-score* de $0,805$, com precisão de 82% e revocação de 79%. É importante lembrar que não é possível comparar diretamente tais resultados, visto que não é possível avaliar a complexidade dos conjuntos de dados utilizados. Um ponto interessante, porém, é que os trabalhos de Ramakrishnan (2011) e Wilson et al. (1996) identificam posições de descanso por meio da análise de posições das mãos que são frequentes nos vídeos. Portanto, a análise é feita individualmente para cada vídeo, diferentemente da proposta deste capítulo, que usa o vídeo A1 para treinamento e o vídeo A2 para validação.

Os experimentos #6, #7 e #8 tiveram por objetivo avaliar o desempenho da tarefa de segmentação de fases considerando as cinco fases possíveis, usando diferentes estratégias para classificação multiclasse. Apesar das estratégias obterem resultados similares, a estratégia hierárquica obteve resultados levemente superiores.

Para a segmentação de fases, uma análise dos erros por *frames* e por segmentos foi realizada para a estratégia hierárquica. A partir dessa análise, é possível avaliar tais resultados em relação aos trabalhos correlatos da Seção 4.2.

Martell e Kroll (2007) considera erros por *frames* e obtém *F-scores* de $0,54$, $0,59$, $0,36$ e $0,67$ para preparação, *stroke*, *hold* e retração, respectivamente. Considerando erros por frames, a abordagem proposta nesta dissertação obtém *F-scores* de $0,55$, $0,75$, $0,33$ e $0,25$. É importante notar, porém, que Martell e Kroll (2007) realiza a segmentação de fases considerando unidades gestuais pré-segmentadas, enquanto a abordagem deste trabalho realiza a segmentação de unidades gestuais de forma automatizada e os erros da segmentação de unidades gestuais prejudicam os resultados das demais classes.

Já Ramakrishnan (2011) considera erros por segmentos, visto que sua abordagem consiste em identificar pontos de transição entre duas fases e classificar esses pontos como o início de uma fase específica. Ou seja, Ramakrishnan (2011) não realiza uma classificação *frame* a *frame*. Ramakrishnan (2011) obtém *F-scores* de $0,743$, $0,799$, $0,953$ e $0,857$ para preparação, *stroke*, *hold* e retração, respectivamente. Ao considerar uma métrica de erros por segmento para a abordagem proposta nesta dissertação, obtém-se *F-scores* de $0,697$, $0,667$, $0,667$ e $0,40$ para preparação, *stroke*, *hold* e retração, respectivamente. Tanto a métrica usada neste trabalho quanto a métrica usada em Ramakrishnan (2011) admitem desvios no início ou no fim das fases (neste trabalho, tais desvios são chamados de erros de transição). Porém, Ramakrishnan (2011) não descreve se existe um limite máximo

para os desvios. Neste trabalho, para o problema de segmentação das fases dos gestos, um segmento é considerado incorreto quando a maior parte dos *frames* no segmento é classificado como pertencente a uma classe diferente da classe real do dado.

Os resultados apresentados neste capítulo consideram o vídeo A1 para treinamento e o A2 para validação, portanto tais resultados não podem ser considerados como resultados gerais para a abordagem. O próximo capítulo visa avaliar os resultados gerais da abordagem, com testes que considerem dependência de usuário e de contexto.

6 Estudo da dependência de usuário e de sessão

No capítulo anterior, foram apresentados modelos criados a partir do vídeo A1 e testados no vídeo A2. Os vídeos foram capturados na mesma sessão com um mesmo usuário, o que significa que é provável que o comportamento do usuário seja o mesmo nos dois vídeos.

Neste capítulo, o objetivo é avaliar como a estratégia proposta se comporta ao considerar diferentes usuários ou mudanças de comportamento dos usuários. Portanto, serão avaliados os resultados de segmentação da unidade gestual (classificação de descanso) e segmentação das fases (classificação multiclasse).

Nesses experimentos, as variáveis obtidas no experimento #1 foram utilizadas: conjuntos não balanceados; com deslocamento igual a 3; velocidade e aceleração escalares das mãos e pulsos; com rótulos na posição central da janela. Para a segmentação da unidade gestual, uma SVM tradicional foi aplicada, com parâmetro de abertura da RBF de 2^{-2} a 2^4, com expoente variando de 1 em 1, e janelas de até 80 *frames*. Para a segmentação das fases dos gestos, a estratégia hierárquica foi usada, com o intervalo do parâmetro de abertura da RBF modificado para 2^2 a 2^5, com expoente variando de 1 em 1, e janelas de até 60 *frames*[1], de acordo com os resultados dos experimentos precedentes.

6.1 Experimento #9: usuário igual, sessão diferente

Neste caso, o vídeo A1 é usado para o treinamento e o vídeo A3 é usado para validação. Os vídeos foram capturados em sessões diferentes, o que implica em diferenças nos vídeos relacionadas, por exemplo, a mudanças de comportamento do usuário em cada sessão. Os vídeos A1 e A2 considerados nos experimentos anteriores foram capturados na mesma sessão, de forma a apresentar maior semelhança entre si.

[1]Para o experimento #12, foram analisadas janelas com até 90 *frames*.

Uma das diferenças que pode ser notada no vídeo A3 é a proporção de *frames* de *hold* no vídeo, superior aos demais vídeos do mesmo usuário, o que pode prejudicar o desempenho da classificação de posições de descanso. A distribuição de classes para cada vídeo é apresentada na Tabela 1, no Capítulo 5.

6.1.1 Segmentação da Unidade Gestual

Em relação à classificação de descanso e gesto, o resultado do modelo com os mesmos parâmetros obtidos no melhor classificador da etapa de testes anteriores foi muito inferior ao resultado obtido para o vídeo A2, como pode ser visto na Tabela 20. O resultado pode indicar que o modelo proposto é altamente sensível à sessão, ou seja, sensível a alterações no comportamento ou humor do usuário. O modelo obtido usa tamanho de janela igual a 44 e parâmetro do RBF igual a 64.

Tabela 20 – *Desempenho do modelo para classificação entre descanso e gesto: treino com A1 e validação com A3.*

Erro	Precisão	Revocação	*F-score*
0,2345	0,6179	0,8412	0,7124

A Tabela 21 mostra uma matriz de confusão baseada nos erros por segmento, conforme a métrica descrita na Seção 5.3.1.3. Considerando essa métrica, nota-se que o modelo treinado a partir do vídeo A1 classifica erroneamente sete (de um total de doze) trechos de gestos como descanso no vídeo A3. Dois trechos de descanso são confundidos com trechos de gestos, porém ambos os trechos classificados incorretamente correspondem a segmentos de descanso pequenos, com apenas nove e treze *frames*.

Tabela 21 – *Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do experimento #9.*

		Classificado como	
		Descanso	Gesto
Classe real	Descanso	10	2
	Gesto	7	5

Ao considerar os erros por *frames*, dos 419 *frames* classificados incorretamente, 85 *frames* são erros de transição, 312 *frames* são erros internos e 22 são frames que correspondem a períodos que foram inteiramente classificados de forma incorreta. Dos 321 *frames* de gesto classificados como descanso, 96 são erros internos que correspondem a *frames* da fase de *hold*, 179 são erros internos que correspondem a *frames* da fase de *stroke*, 19 são erros de transição e os 18 frames restantes são erros internos que correspondem a

frames das fases de preparação e retração. Dos 98 frames de descanso classificado como gesto, 66 são erros de transição, 10 são erros internos e 22 correspondem a dois segmentos de descanso pequenos.

Comparando trechos dos sinais da velocidade da mão direita dos dois vídeos – Figura 15 (A1) e Figura 16 (A3), nota-se que: o sinal do vídeo A1 apresenta maior variação na velocidade nos períodos de descanso, porém com velocidade menor que nos períodos de gesto do mesmo vídeo; e o sinal do vídeo A3 apresenta períodos de gesto, mais especificamente *stroke*, com pouca variação na velocidade e velocidade baixa. Assim, é compreensível que o modelo considere períodos com velocidade mais baixa, ainda que com certa variação, como descanso.

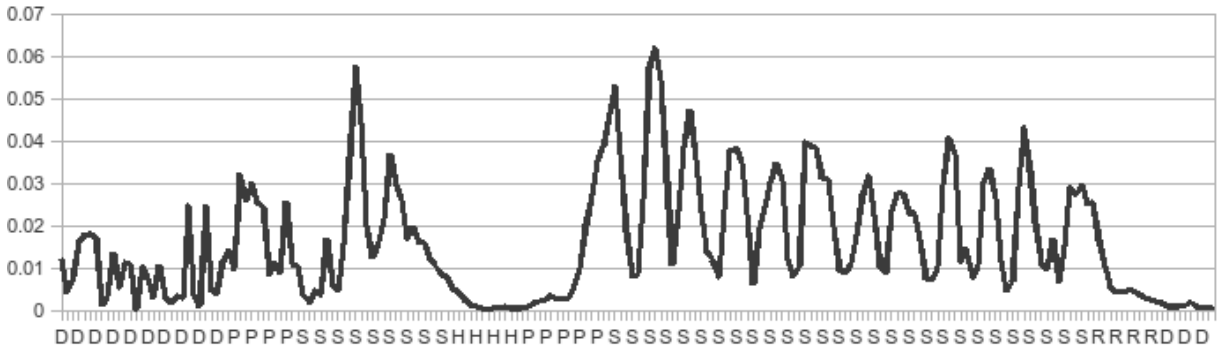

Figura 15 *– Sinal da velocidade da mão direita do vídeo A1.*

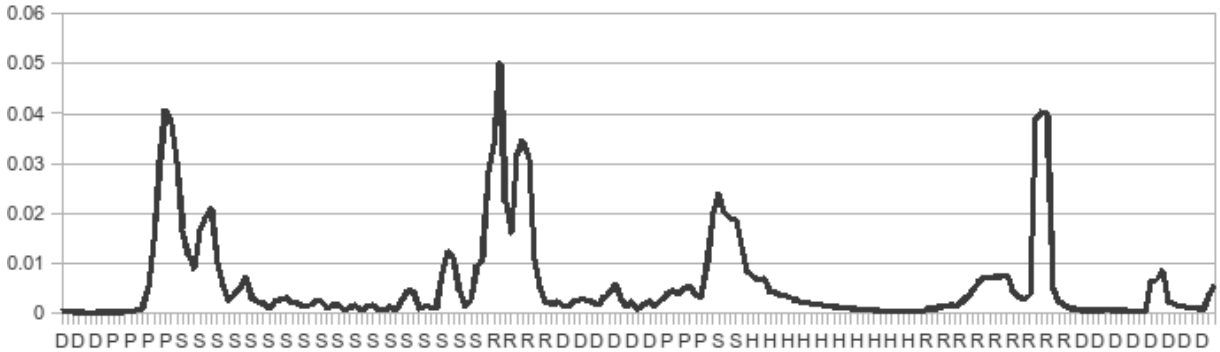

Figura 16 *– Sinal da velocidade da mão direita do vídeo A3.*

6.1.2 Segmentação das Fases

O desempenho da estratégia hierárquica para o vídeo A3 é consideravelmente inferior ao obtido com o vídeo A2. É possível perceber uma queda no *F-score* para todas as classes, conforme apresentado na Tabela 22. O melhor conjunto de classificadores obtido neste experimento apresenta um erro de classificação de $43,39\%$.

Neste caso, o classificador de descanso considera uma janela de 36 frames com parâmetro de RBF igual a 16; o classificador de *hold* considera uma janela de 15 frames com parâmetro de RBF igual a 16; o classificador de *stroke* considera uma janela de 37 frames com parâmetro de RBF igual a 32; e o classificador de preparação/retração considera uma janela de 25 frames com parâmetro de RBF igual a 16.

A Tabela 23 apresenta o desempenho de cada classificador analisado de forma isolada, sem considerar os erros provenientes dos classificadores anteriores acrescentados pela estratégia hierárquica.

Tabela 22 – *Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	62,02	85,11	0,7175
Preparação	59,43	45,16	0,5132
Stroke	49,35	57,01	0,5291
Hold	67,65	15,33	0,2500
Retração	34,92	10,58	0,1624
Média	54,67	42,64	0,4344

Tabela 23 – *Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	60,48	84,27	0,7042
Preparação	59,43	88,73	0,7119
Stroke	49,35	86,65	0,6289
Hold	67,65	48,94	0,5679

Observando a matriz de confusão na Tabela 24, nota-se o impacto do maior número de *frames* de *hold* no desempenho do modelo. Devido à semelhança das classes descanso e *hold* e às diferenças entre os padrões do vídeo A1 e do vídeo A2, $45,8\%$ dos *frames* de *hold* foram classificados como descanso, prejudicando a precisão para a classe descanso e a revocação para a classe *hold*.

Analisando os erros por segmento, conforme descrito na Seção 5.3.8, obtém-se a matriz de confusão apresentada na Tabela 25. No total, 35 segmentos foram classificados incorretamente de um total de 68. Desses 35 segmentos, 27 são segmentos inferiores a um segundo (30 *frames*) e, por esse motivo, mais propícios a serem prejudicados por erros de transição.

O tamanho dos segmentos também pode ser um fator que dificulta a identificação dos segmentos de preparação, *hold* e retração: esses segmentos são mais curtos do que

Tabela 24 – *Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**560**	12	76	8	2
	P	27	**126**	107	3	16
Classe real	S	183	29	**305**	0	18
	H	103	1	18	**23**	5
	R	30	44	112	0	**22**

Tabela 25 – *Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**11**	0	2	0	0
	P	1	**10**	7	0	0
Classe real	S	5	1	**11**	0	1
	H	4	0	2	**1**	0
	R	2	2	8	0	**0**

segmentos de descanso e *stroke*. No vídeo A3, por exemplo, segmentos de preparação, *hold* e retração tem, em média, 16, 21 e 17 *frames*, enquanto segmentos de descanso e *stroke* tem, em média, 51 e 30 *frames*.

6.2 Experimento #10: história igual, usuário diferente

O segundo teste visa avaliar a independência de usuário da abordagem proposta. Neste caso, os modelos foram treinados com o vídeo A1 e testados com o vídeo B1, ou seja, os dois vídeos continham dois usuários diferentes, porém contando a mesma história.

6.2.1 Segmentação da Unidade Gestual

Como pode ser visto na Tabela 26, os resultados para a classificação entre descanso neste caso são comparáveis aos testes dependentes de usuário. O modelo com melhor desempenho considera uma janela de 14 *frames* e parâmetro da RBF igual a 4.

É importante notar que existem diferenças na forma de gesticular dos usuários envolvidos, principalmente em relação à distribuição das classes em cada vídeo. No vídeo A1, 40% dos *frames* correspondem a posições de descanso, enquanto no vídeo B1 a proporção

Tabela 26 – *Desempenho do modelo para classificação entre descanso e gesto: treino com A1 e validação com B1.*

Erro	Precisão	Revocação	*F-score*
0,1480	0,6708	0,8030	0,7310

de posições de descanso é de $25,7\%$. A proporção de períodos de *hold*, que podem ser facilmente confundidos com descanso, também é diferente: A1 tem menos de 3% de *frames* de *hold*, enquanto B1 tem $17,4\%$.

Porém, apesar da existência de diferenças na forma de gesticular dos usuários envolvidos, o modelo foi capaz de obter um resultado similar ao obtido quando um mesmo usuário se comporta de maneira diferente. Comparando os resultados dos experimentos #9 e #10 nota-se que os resultados indicam que a estratégia depende do comportamento de determinado usuário em determinado momento, e não apenas do usuário.

Analisando os erros por *frames*, dos 159 *frames* classificados incorretamente, 44 *frames* são erros de transição, 99 *frames* são erros internos e 16 são frames que correspondem a períodos que foram inteiramente classificados de forma incorreta. Dos 106 *frames* de gesto classificados como descanso, 71 são erros internos que correspondem a *frames* da fase de *hold*, 8 são erros internos que correspondem a *frames* da fase de *stroke*, 16 são erros de transição e os 11 frames restantes são erros internos que correspondem a *frames* das fases de preparação e retração. Dos 53 frames de descanso classificado como gesto, 28 são erros de transição, 9 são erros internos e 16 correspondem a dois segmentos de descanso pequenos.

A Tabela 27 mostra que 40% dos segmentos de descanso foram classificados como unidade gestual. Porém, é interessante notar que, desses quatro períodos classificados incorretamente, três correspondem a períodos curtos de descanso – com nove, sete e quatro *frames* –, nos quais pode haver divergências sobre a classificação mesmo por especialistas. Já no caso de unidades gestuais classificadas como descanso, apenas uma unidade foi classificada erroneamente. Analisando os erros da unidade gestual, nota-se que os erros são causados apenas pela confusão com *frames* de *hold*.

Tabela 27 – *Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do experimento #10.*

		Classificado como	
		Descanso	Gesto
Classe real	Descanso	6	4
	Gesto	1	8

6.2.2 Segmentação das Fases

A estratégia hierárquica para classificação multiclasse aplicada ao vídeo B1 obtém resultados superiores que o experimento #9 para as classes de descanso, *stroke* e retração, e inferiores para as demais classes, como pode ser visto na Tabela 29. Foram utilizados quatro classificadores: de descanso, com 14 frames na janela e parâmetro RBF igual a 4; *hold*, com 6 frames na janela e parâmetro RBF igual a 4; stroke, com 28 frames na janela e parâmetro RBF igual a 8; e preparação/retração, com 28 frames na janela e parâmetro RBF igual a 32.

A matriz de confusão na Tabela 28 mostra que os principais motivos para os erros da abordagem estão relacionados à confusão entre descanso e *hold* e à confusão entre preparação e retração, correspondendo a $38,5\%$ dos erros da estratégia. Além disso, como apresentado na Tabela 29, a estratégia não é capaz de identificar a classe *hold* de forma satisfatória: os erros relacionados à baixa capacidade de revocação da estratégia para a classe *hold* correspondem a 34% dos erros. A abordagem obtém $44,71\%$ de erro de classificação.

Tabela 28 *– Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**229**	14	17	1	21
	P	24	**98**	39	5	66
Classe real	S	8	32	**197**	4	11
	H	71	40	4	**26**	50
	R	3	49	24	3	**51**

Tabela 29 *– Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	68,36	81,21	0,7423
Preparação	42,06	42,24	0,4215
Stroke	70,11	78,17	0,7392
Hold	66,67	13,61	0,2261
Retração	25,63	39,23	0,3100
Média	54,56	50,89	0,4878

A Tabela 30 apresenta o desempenho de cada classificador analisado de forma isolada, sem considerar os erros provenientes dos classificadores anteriores acrescentados pela estratégia hierárquica.

Tabela 30 – *Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	67,08	80,30	0,7310
Preparação	42,06	59,76	0,4937
Stroke	71,38	82,08	0,7636
Hold	67,67	21,67	0,3270

Analisando os erros por segmento, é possível verificar na Tabela 31 que 19% dos erros de segmento correspondem a segmentos de *hold* classificados como descanso e 35% correspondem a segmentos classificados erroneamente como retração. Novamente, dos 31 segmentos incorretos, apenas três segmentos têm mais de 30 *frames*, e esses tem apenas 35, 35 e 34 *frames*.

Tabela 31 – *Desempenho do melhor conjunto de classificadores usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**8**	0	0	0	2
	P	0	**8**	4	0	3
Classe real	S	0	2	**5**	0	1
	H	5	1	1	**0**	3
	R	0	3	1	0	**4**

6.3 Experimento #11: misturando usuários, sessões e histórias

O terceiro teste consiste em treinar com os vídeos A1 e B1 e validar com os vídeos A3 e B3. O objetivo neste caso é avaliar a capacidade de generalização da abordagem considerando diferentes estilos de gesticulação.

6.3.1 Segmentação da Unidade Gestual

Neste caso, o desempenho do modelo para classificar descanso e gesto foi inferior aos resultados dos demais testes, indicando que talvez seja mais adequado considerar uma abordagem dependente de usuário (e até mesmo dependente do comportamento do usuário), conforme pode ser visto na Tabela 32. O melhor modelo obtido para os vídeos A3 e B3 considera 42 frames na janela e parâmetro da RBF igual a 8.

Tabela 32 – *Desempenho do modelo para classificação entre descanso e gesto: treino com A1+B1 e validação com A3+B3.*

Erro	Precisão	Revocação	*F-score*
0,1852	0,6133	0,7989	0,6939

Ao analisar os erros por segmento, temos 12 segmentos de descanso e 12 segmentos de gesto para o vídeo A3, e 10 segmentos de descanso e 10 segmentos de gesto para o vídeo B3. Para o vídeo A3, o novo classificador melhora um pouco o reconhecimento de gestos, mas piora o desempenho na identificação do descanso, em relação aos resultados obtidos no experimento #9, como mostra a matriz de confusão na Tabela 33.

Tabela 33 – *Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do vídeo A3 no experimento #11.*

		Classificado como	
		Descanso	Gesto
Classe real	Descanso	8	4
	Gesto	4	8

Já para o vídeo B3, o classificador tem um bom desempenho, como mostra a matriz de confusão na Tabela 34. Apenas um segmento de descanso é classificado como gesto, e é um segmento curto, com 8 *frames*. Dos 10 segmentos de gestos, três foram classificados como descanso.

Tabela 34 – *Matriz de confusão baseada nos erros de segmento para a segmentação de unidade gestual do vídeo B3 no experimento #11.*

		Classificado como	
		Descanso	Gesto
Classe real	Descanso	9	1
	Gesto	7	3

É interessante notar que esse classificador foi capaz de identificar alguns segmentos de descanso bem curtos – dois com 5 *frames* e um com 16 *frames* –, ainda que com pequenos deslocamentos no início e no final do segmento.

Ao analisar os erros por *frames* considerando os dois vídeos, dos 658 *frames* classificados incorretamente, 153 *frames* são erros de transição, 456 *frames* são erros internos e 49 são *frames* que correspondem a períodos que foram inteiramente classificados de forma incorreta.

Dos 470 *frames* de gesto classificados como descanso, 162 são erros internos que correspondem a *frames* da fase de *hold*, 239 são erros internos que correspondem a *frames*

da fase de *stroke*, 29 são erros de transição e os 40 frames restantes são erros internos que correspondem a *frames* das fases de preparação e retração. Ou seja, $34,4\%$ dos *frames* de unidades gestuais classificados incorretamente correspondem diretamente a *frames* de *hold*.

Dos 188 *frames* de descanso classificados como gesto, 124 são erros de transição, 15 são erros internos e 49 correspondem a quatro segmentos de descanso pequenos – com 8, 9, 13 e 19 *frames* –inteiramente classificados como gesto.

6.3.2 Segmentação das Fases

Neste teste, foram gerados dois conjuntos de classificadores: ambos usaram os vídeos A1 e B1 para treinamento, mas cada conjunto considera os melhores classificadores para cada vídeo considerado na validação – A3 e B3.

O primeiro conjunto de classificadores, aplicado ao vídeo A3, é composto por um classificador de descanso, com 31 frames na janela e parâmetro RBF igual a 4; um classificador de *hold*, com 18 frames na janela e parâmetro RBF igual a 16; um classificador de *stroke*, com 31 frames na janela e parâmetro RBF igual a 8; e um classificador de preparação/retração, com 58 frames na janela e parâmetro RBF igual a 32.

O segundo conjunto de classificadores, aplicado ao vídeo B3, é composto por um classificador de descanso, com 22 frames na janela e parâmetro RBF igual a 4; um classificador de *hold*, com 60 frames na janela e parâmetro RBF igual a 16; um classificador de *stroke*, com 37 frames na janela e parâmetro RBF igual a 8; e um classificador de preparação/retração, com 57 frames na janela e parâmetro RBF igual a 16.

Para avaliar a abordagem de forma justa, o vídeo A3 foi analisado pelo conjunto de classificadores com melhores resultados para o vídeo C3 e o vídeo C3 foi analisado pelo conjunto de classificadores com melhores resultados para o vídeo A3. Por fim, o conjunto que obteve o melhor resultado considerando os dois vídeos – neste caso, o conjunto obtido com os melhores parâmetros encontrados para o vídeo A3, ou seja, o primeiro conjunto de classificadores – é avaliado neste experimento.

A Tabela 35 apresenta os resultados obtidos pelo primeiro conjunto de classificadores para os vídeos A3 e B3. Na validação com o vídeo A3, $45,79\%$ dos *frames* são classificados incorretamente. Para o vídeo B3, o erro de classificação corresponde a $46,12\%$ dos *frames*.

A Tabela 36 apresenta a matriz de confusão considerando os resultados do vídeo A3.

Para o vídeo B3, nota-se que os resultados para as classes de descanso e *hold* foram inferiores em relação aos obtidos para o vídeo A3, porém são superiores para as classes de preparação, *stroke* e retração. A Tabela 37 apresenta a matriz de confusão considerando os resultados do vídeo B3.

Tabela 35 – *Desempenho do primeiro conjunto de classificadores para os vídeos A3 e B3 usando a estratégia hierárquica – precisão (em %), revocação (em %) e F-score.*

	Vídeo A3			Vídeo B3		
Classe	Precisão	Revocação	*F-score*	Precisão	Revocação	*F-score*
Descanso	65,89	77,51	0,7123	61,18	66,48	0,6372
Preparação	47,25	52,33	0,4966	58,33	49,70	0,5367
Stroke	52,28	42,80	0,4707	56,04	63,24	0,5942
Hold	51,25	27,33	0,3565	21,88	24,31	0,2303
Retração	28,82	31,73	0,3021	54,17	42,33	0,4752
Média	49,10	46,34	0,4676	50,32	49,21	0,4947

Tabela 36 – *Desempenho do primeiro conjunto de classificadores para o vídeo A3 usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**510**	23	58	29	38
	P	20	**146**	85	6	22
Classe real	S	151	54	**229**	4	97
	H	81	12	10	**41**	6
	R	12	74	56	0	**66**

Tabela 37 – *Desempenho do primeiro conjunto de classificadores para o vídeo B3 usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**238**	17	16	54	33
	P	20	**168**	100	29	21
Classe real	S	65	42	**246**	20	16
	H	64	13	25	**35**	7
	R	2	48	52	22	**91**

A Tabela 38 apresenta o desempenho de cada classificador analisado isoladamente, para o conjunto de classificadores com parâmetros ajustados para o vídeo A3.

Ao analisar os erros por segmento para os mesmos vídeos, nota-se que existem algumas confusões recorrentes: *hold* classificado como descanso; retração classificada como preparação; e preparação classificada como *stroke*. Problemas relacionados à precisão do

Tabela 38 *– Desempenho de cada classificador do primeiro conjunto analisado isoladamente para o vídeo A3 – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	64,52	76,43	0,6997
Preparação	48,83	86,90	0,6253
Stroke	52,28	60,26	0,5599
Hold	51,25	59,42	0,5503

classificador de descanso também são responsáveis por $30,4\%$ dos erros, porém é importante notar que, dos 17 segmentos classificados erroneamente como descanso, oito são períodos de hold – fase muito semelhante ao descanso em termos de velocidade e aceleração; e quatro são períodos de preparação, que, na maior parte dos casos, consistem em erros de transição.

Tabela 39 *– Desempenho do primeiro conjunto de classificadores para o vídeo A3 usando a estratégia hierárquica – matriz de confusão.*

		Classificado como				
		D	P	S	H	R
	D	**11**	0	0	0	2
	P	2	**13**	3	0	0
Classe real	S	3	1	**9**	0	5
	H	4	1	0	**1**	1
	R	0	5	2	0	**5**

Tabela 40 *– Desempenho do primeiro conjunto de classificadores para o vídeo B3 usando a estratégia hierárquica – matriz de confusão dos erros por segmento.*

		Classificado como				
		D	P	S	H	R
	D	**9**	0	1	0	1
	P	2	**10**	2	0	2
Classe real	S	2	1	**9**	0	1
	H	4	0	3	**2**	1
	R	0	3	3	4	**4**

6.4 Experimento #12: treinando e testando com o mesmo vídeo

Dada à clara dependência de usuário e de sessão, a abordagem proposta poderia ser usada para facilitar a segmentação de vídeos maiores. Neste caso, o usuário poderia usar um trecho inicial do vídeo para treinamento e usar o modelo treinado para obter a segmentação do restante do vídeo.

6.4.1 Segmentação da Unidade Gestual

Para testar essa opção, 30% do vídeo A1 foi usado para treinamento – ou seja, cerca de 20 segundos de vídeo – e o restante foi usado para validação. A Tabela 41 mostra os resultados para o melhor modelo gerado para classificar descanso e gesto, com uma janela de 25 *frames* e parâmetro RBF igual a 8. Apesar da quantidade reduzida de *frames* usados no treinamento (535 *frames*), o modelo foi capaz de obter bons resultados para os dados de validação.

Tabela 41 – *Desempenho do modelo para classificação entre descanso e gesto: treino com A1 (30%) e validação com A1 (70%).*

Erro	Precisão	Revocação	*F-score*
0,1080	0,8129	0,9072	0,8575

Ao considerar erros por segmento, os dados de validação incluem sete segmentos de descanso e sete segmentos de gesto. Neste teste, um segmento de cada fase foi classificado incorretamente. A Tabela 42 apresenta os detalhes dos dois segmentos classificados incorretamente. É possível perceber que o período de descanso classificado como gesto é um período curto, com apenas 11 *frames*, dos quais 5 foram classificados incorretamente. No caso do gesto classificado incorretamente, um período composto principalmente de *strokes* foi confundido com um período de descanso, possivelmente por se tratar de um movimento repetitivo em uma pequena área, podendo acarretar em velocidades menores devido à menor amplitude dos movimentos.

Tabela 42 – *Erros do modelo para os segmentos de descanso e gesto: treino com A1 (30%) e validação com A1 (70%).*

Fase	Classificado como	Tamanho do segmento	Erros de transição	Erros internos
Descanso	Gesto	11	5	0
Gesto	Descanso	96	0	58

Para avaliar a estratégia, um novo teste foi realizado considerando 70% dos dados para treinamento e 30% para validação. O objetivo é avaliar se a queda de desempenho em relação ao experimento #1 se deve apenas à quantidade reduzida de dados. Os resultados, apresentados na Tabela 43, mostram que, com uma maior quantidade de dados, é possível obter melhores resultados. O melhor resultado foi obtido com uma janela de 17 *frames* e parâmetro da RBF igual a 32. Neste caso, nenhum segmento foi classificado incorretamente, incluindo o segmento curto de descanso com apenas 11 *frames*, que foi encontrado com um deslocamento de quatro *frames* para frente. Portanto, a abordagem de treinar com o início do vídeo para classificar o restante parece promissora.

Tabela 43 – *Desempenho do modelo para classificação entre descanso e gesto: treino com A1 (70%) e validação com A1 (30%).*

Erro	Precisão	Revocação	*F-score*
0,0790	0,8745	0,9664	0,9182

Analisando os erros por *frames* desse segundo teste, dos 41 *frames* classificados incorretamente, 24 *frames* são erros de transição e 17 *frames* são erros internos. Dos 33 *frames* de gesto classificados como descanso, 16 são erros internos e 17 são erros de transição. Dos 8 *frames* de descanso classificado como gesto, 7 são erros de transição e um *frame* corresponde a um erro interno.

Considerando que a estratégia proposta nesse teste, é dependente de cada vídeo analisado, é mais adequado comparar esse resultado aos trabalhos correlatos apresentados na Seção 4.2. Neste caso, os resultados são muito similares aos obtidos por Ramakrishnan (2011) – *F-score* de $0,913$, com precisão de 87% e revocação de 96% – e consideravalmente superiores aos resultados obtidos por Wilson et al. (1996) – *F-score* de $0,805$, com precisão de 82% e revocação de 79%.

6.4.2 Segmentação das Fases

Neste teste, cada classificador tem um número diferente de *frames* na janela e um parâmetro de RBF diferente. O classificador de descanso tem 17 *frames* na janela e parâmetro da RBF igual a 32, o de *stroke* tem 32 *frames* e parâmetro da RBF igual a 8, e o de preparação/retração tem 71 *frames* e parâmetro da RBF igual a 8. Não foi um escolhido um classificador de *hold* neste teste porque os últimos 30% do vídeo avaliado não continham *frames* de *hold*. Os resultados da Tabela 44 mostram resultados superiores a todos os resultados obtidos para a segmentação das fases do gesto, indicando que a abordagem de treinar com início do vídeo visando rotular o final do vídeo pode ser a melhor opção para resolver o problema.

Tabela 44 – *Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	87,82	96,75	0.9207
Preparação	87,50	35,00	0.5000
Stroke	84,11	67,20	0.7471
Retração	39,33	67,31	0.4965
Média	74,69	66,56	0.6661

A Tabela 45 apresenta o desempenho de cada classificador analisado de forma isolada,

sem considerar os erros provenientes dos classificadores anteriores acrescentados pela estratégia hierárquica.

Tabela 45 – *Desempenho de cada classificador da estratégia hierárquica analisado isoladamente – precisão (em %), revocação (em %) e F-score.*

Classe	Precisão	Revocação	*F-score*
Descanso	87,45	96,64	0,9182
Preparação	87,50	66,67	0,7568
Stroke	84,11	74,71	0,7913

É possível notar bons desempenhos para as classes descanso e *stroke*. Novamente, as classes preparação e retração apresentam resultados inferiores. A matriz de confusão é apresentada na Tabela 46. A abordagem proposta obtém $21,44\%$ de erro.

Tabela 46 – *Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – matriz de confusão.*

		Classificado como			
		D	P	S	R
	D	**238**	2	2	4
	P	5	**14**	14	7
Classe real	S	19	0	**127**	43
	R	9	0	8	**35**

Ao analisar os erros por segmento, obtém-se a matriz de confusão apresentada na Tabela 47. Por meio da matriz, é possível observar que, de um total de 16 segmentos, cinco são classificados incorretamente. Uma breve descrição dos quatro segmentos é apresentada na Tabela 48.

Tabela 47 – *Desempenho do melhor conjunto de classificadores usando a estratégia um contra todos – matriz de confusão dos erros por segmento.*

		Classificado como			
		D	P	S	R
	D	**3**	0	0	0
	P	1	**2**	0	1
Classe real	S	1	0	**3**	1
	R	1	0	0	**2**

É interessante notar que dos cinco segmentos classificados incorretamente, quatro são segmentos curtos – com tamanho inferior a 15 *frames* – que não foram identificados, de forma que os *frames* foram classificados de acordo com a classe atribuída ao segmento imediatamente anterior ou posterior. No caso do segmento de 34 *frames*, correspondente a um período de *stroke*, o início do segmento foi classificado corretamente. Após 10 *frames*,

Tabela 48 – *Erros do modelo para os segmentos de fases: treino com A1 (70%) e validação com A1 (30%).*

Fase	Classificado como	Tamanho do segmento	*Frames* incorretos
Preparação	Descanso	6	6
Preparação	Retração	14	14
Stroke	Retração	34	24
Retração	Descanso	8	8
Retração	Stroke	7	7

o classificador deslocou as fases de retração e descanso que viriam após o segmento, apontando 15 *frames* de retração e 9 *frames* de descanso dentro do segmento correspondente ao período de *stroke*.

6.5 Considerações Finais

Este capítulo apresentou um estudo sobre a dependência de usuário e de sessão para a abordagem proposta. Os resultados indicam que existe uma dependência relacionada ao comportamento do usuário, que pode se modificar dependendo de fatores adversos, como o próprio humor do usuário.

Da mesma forma, os resultados indicam, inicialmente, que não é adequado investir na construção de um modelo único para diversos usuários, visto que até mesmo os parâmetros ideais dos classificadores podem mudar para vídeos diferentes.

Portanto, propõe-se adotar a segmentação manual de um trecho inicial do vídeo, visando realizar a segmentação automática do restante do vídeo. Essa abordagem não automatiza todo o processo, porém garante certa homogeneidade entre o padrão de treinamento e o padrão de validação, proporcionando resultados mais confiáveis. Nesse caso, constrói-se um modelo específico para cada vídeo, de forma a evitar a existência de modificações bruscas no comportamento do usuário.

7 Conclusões

O presente trabalho teve por objetivo desenvolver estratégias para realizar a segmentação das unidades gestuais e das fases dos gestos de forma automatizada, visando dar suporte à interpretação do papel da gestualidade na constituição do discurso. A identificação de cada fase do gesto foi tratada como uma tarefa de classificação supervisionada, utilizando Máquinas de Vetores Suporte. Para melhor adequar a SVM à natureza temporal do problema, abordagens que consideram a dependência temporal dos dados foram investigadas.

Para atingir tal objetivo, foram realizados: um estudo sobre técnicas de SVM e suas variações que incorporam raciocínio temporal; um estudo sobre representação temporal; um estudo sobre os trabalhos correlatos na área de análise de gestos; a coleta de dados usando Microsoft Kinect; e o desenvolvimento de algoritmos para a extração de características e para a realização dos experimentos. Em relação aos experimentos e à análise de resultados, o trabalho foi realizado em três etapas: segmentação das unidades gestuais; segmentação das fases do gesto; e avaliação dos resultados considerando dependência de história, de usuário e de sessão.

Para a segmentação das unidades gestuais, diversas estratégias foram consideradas – incluindo o uso de janelas e representação com características no domínio do tempo, no domínio da frequência e no domínio de tempo-frequência – e submetidas à análise de Máquinas de Vetores Suporte tradicionais. Além disso, raciocínio temporal incorporado à SVM foi analisado sob experimentos com *kernel* recursivo e *Support Vector Echo-State Machine*. Nessa etapa, os vídeos usados para treinamento e teste dos modelos classificadores são histórias diferentes contadas por um mesmo usuário.

O objetivo dessa etapa foi avaliar os métodos de classificação e as formas de representação mais adequadas para o problema. Em relação aos métodos de classificação, foi possível perceber que as Máquinas de Vetores Suporte incorporando raciocínio temporal não superaram o desempenho da Máquina de Vetores Suporte tradicional. Da mesma

forma, a extração de características nos domínios do tempo, frequência e tempo-frequência não possibilitaram desempenhos superiores aos obtidos com o janelamento simples. Por fim, o melhor resultado foi obtido com uma Máquina de Vetores Suporte tradicional e dados janelados considerando o *frame* de interesse no meio da janela, com *frames* representados pelas medidas de velocidade e aceleração das mãos e pulsos. O modelo gerado – com erro de $9,83\%$ e *F-score* de $0,8774$ – indica um grande potencial para aplicação na área de análise de gestos, funcionando como uma triagem inicial dos *frames* de descanso e unidade gestual.

Para a segmentação das fases do gesto, o melhor método de classificação e a melhor forma de representação para a segmentação das unidades gestuais foi usado para a segmentação das fases do gesto considerando os mesmos vídeos da primeira etapa. Três formas diferentes foram aplicadas para realizar a classificação multiclasse: estratégia um contra um, estratégia um contra todos e estratégia hierárquica. Nessa etapa, foi possível perceber que a estratégia hierárquica proposta sacrifica a medida de revocação em prol de um aumento da medida de precisão. O resultado final, porém, indica que o aumento na precisão compensa a redução na revocação, produzindo resultados melhores que as demais estratégias em termos de *F-score*. Embora os resultados para as classes de preparação, *hold* e retração não sejam tão bons quanto os resultados para as fases de descanso e *stroke*, análises mais detalhadas sobre as situações em que os erros ocorrem nos permitem avaliar que, em muitos casos, trata-se de uma limitação da abordagem relacionada ao tipo de representação usada. Por exemplo, é difícil distinguir as fases de *hold* e descanso em relação a velocidade e aceleração apenas.

No estudo da dependência de história, de usuário e de sessão, foi explorada, com mais detalhes e variabilidade de configuração de experimentos, a melhor combinação de método, representação e estratégia para classificação multiclasse encontrada para a segmentação das unidades gestuais e para a segmentação das fases do gesto nas etapas anteriores. Assim, nesses novos experimentos, tal combinação foi usada para avaliar outros vídeos, considerando dependência de história, de sessão e de usuário. A partir desses experimentos, concluiu-se que a abordagem é adequada para aplicação na classificação de um mesmo vídeo, usando o começo do vídeo para treinamento, visando automatizar a segmentação do restante do vídeo. Contudo, ela tem um desempenho reduzido quando a complexidade do problema aumenta ao considerar vídeos de usuários diferentes ou mesmo vídeos diferentes de um mesmo usuário gravados em sessões diferentes.

As próximas sessões apresentam um resumo das principais contribuições do trabalho e

também das sugestões de extensões, que tratam de cobrir lacunas que não foram possíveis de serem tratadas aqui.

7.1 Principais Contribuições

Além da proposição de estratégias para a segmentação das fases dos gestos, as principais contribuições geradas a partir deste trabalho incluem:

- Revisão sistemática sobre análise automatizada de gestos, focando nos aspectos temporais da análise, disponível sob a forma de relatório técnico em Madeo e Peres (2012).
- Revisão sistemática sobre SVM com raciocínio e representação temporal, publicada sob a forma de dois artigos: Madeo et al. (2012a), cobrindo SVM com representação temporal; e Madeo et al. (2012b), cobrindo SVM com raciocínio temporal.
- Conjunto de dados composto por seis vídeos para o problema de segmentação de fases dos gestos sob o contexto de contação de histórias, com duas rotulações diferentes para cada vídeo.
- Estudo de representação e raciocínio temporal no problema de análise de gestos.
- Proposição da estratégia para segmentação das fases dos gestos.
- Estudo referente à dependência de usuário, sessão e história, no contexto da estratégia de segmentação das fases dos gestos proposta.
- Publicação de resultados com experimentos pilotos, usando uma representação não refinada para velocidade e aceleração escalares (MADEO et al., 2013).

7.2 Trabalhos Futuros

Durante o desenvolvimento deste projeto, algumas ideias foram levantadas para trabalhos futuros, visando tanto aprimorar o desempenho das estratégias desenvolvidas, quanto cobrir alguns pontos que ficaram em aberto. As principais extensões levantadas são listadas a seguir:

- **Representação dos dados:**

  Neste trabalho, as medidas de velocidade e aceleração foram consideradas para representar cada *frame*. Pesquisadores da área de estudos dos gestos, porém, argumentam que outras características poderiam ser importantes para a segmentação das fases dos gestos. A trajetória da mão, dada pela sequência de posições da mão durante o movimento, pode ser útil para diferenciar fases de preparação e retração, por exemplo. A configuração das mãos também pode ser usada para determinar as transições entre as fases com maior precisão. A velocidade angular das mãos em relação aos pulsos pode ser útil para identificar movimentos internos da mão, ajudando a identificar certos tipos de *stroke* e o fim da fase de preparação. Trabalhos correlatos, como Ramakrishnan (2011) e Wilson et al. (1996), consideram apenas a posição das mãos para determinar posições de descanso. Outra sugestão dos pesquisadores da área de estudos de gestos é considerar uma métrica que avalie a tensão das mãos em um determinado *frame*, já que a tensão ajudaria a distinguir posição de descanso da fase de *hold*. Além disso, é preciso realizar um estudo mais detalhado sobre as características nos domínios do tempo, frequência e tempo-frequência, discutindo também a influência da frequência de aquisição dos dados no desempenho dos modelos.

- **Diferentes contextos, usuários e sessões:**

  Neste trabalho, as estratégias desenvolvidas foram aplicadas no contexto de contação de histórias, considerando dois usuários distintos. Foram realizados experimentos relativos à dependência da história contada, de usuário e à variação de comportamento de um mesmo usuário em sessões diferentes. Tanto do ponto de vista computacional quanto do ponto de vista de aplicação, seria interessante aplicar as estratégias desenvolvidas: em outros contextos, como entrevistas e debates; considerando maior número de usuários; e considerando diferentes sessões para cada usuário, incluindo mudanças de humor ou situações nas quais o usuário se sinta mais ou menos confortável. Do ponto de vista computacional, esses testes ajudariam a confirmar se a estratégia de fato precisa ser aplicada a cada vídeo individualmente ou se é possível criar modelos para um único usuário (independente de contexto) ou para um único contexto (independente de usuário) considerando um conjunto de dados maior. Do ponto de vista dos estudos dos gestos, os testes forneceriam material para os pesquisadores avaliarem diferenças de comportamento gestual de acordo com as variáveis consideradas. Também é possível realizar todos os experimentos considerando língua de sinais, ao invés de considerar apenas a gesticulação que acompanha a língua oral.

- **Análises sobre a quantidade necessária de dados de treinamento:**

  Como apresentado no Capítulo 6, esta pesquisa indicou que a segmentação de fases de gestos aqui proposta deve ser treinada para cada vídeo processado. Ou seja, uma parte inicial do vídeo deve ser rotulada manualmente para que se possa treinar um modelo capaz de segmentar de forma automatizada o restante do vídeo. Porém, é necessário realizar uma investigação sobre a quantidade de dados necessária para criar o modelo. Também é preciso avaliar se a quantidade de dados muda dependendo do usuário, do contexto e até mesmo se a quantidade varia para cada vídeo.

- **Tratar cada mão individualmente:**

  Novas linhas de pesquisa na área de estudos de gestos defendem que cada mão deve ser avaliada individualmente. Ou seja, uma mão pode estar em descanso enquanto a outra executa um gesto, por exemplo. Portanto, seria interessante criar estratégias para segmentar as fases de cada mão de forma individual. Nesse caso, seria preciso avaliar: se existem diferenças nos padrões das fases da mão dominante e da mão não dominante; se elas precisam ser tratadas separadamente ou se é possível criar um modelo único de segmentação que sirva para as duas mãos; e se é necessário utilizar a informação das duas mãos para avaliar a fase de cada uma das mãos, como indica o trabalho de Bryll et al. (2001) que usa uma relação entre a quantidade de movimento das duas mãos para determinar se cada uma das mãos está em *hold*.

- **Divergência entre especialistas:**

  Seria interessante avaliar os resultados considerando um maior número de segmentações feitas por especialistas humanos, visando entender a relação entre a divergência de diferentes especialistas com a subjetividade inerente ao problema, e então estudar como essa relação deve ser considerada na avaliação das estratégias automatizadas.

- **Gramática para segmentação das fases dos gestos:**

  A segmentação das fases dos gestos como um todo não obteve resultados satisfatórios na maior parte dos experimentos devido ao desempenho ruim para a classificação de algumas classes, especialmente preparação e *hold*. Nesse caso, seria possível classificar as classes de descanso e *stroke*, e aplicar uma gramática para identificar as demais classes de acordo com sua posição em relação aos períodos de descanso e *stroke*.

7.3 Considerações Finais

A automação de tarefas na área de estudos dos gestos compõe uma área de pesquisa recente em computação e a quantidade de trabalhos visando a segmentação das fases dos gestos ainda é pequena. Portanto, este trabalho apresenta um estudo inicial sobre a segmentação das fases dos gestos usando Máquinas de Vetores Suporte, e ainda existe muito trabalho a ser feito.

Uma das limitações deste trabalho é o uso do dispositivo Microsoft Kinect para obtenção dos dados. Do ponto de vista da área de estudos de gestos, é interessante que se possa realizar o rastreamento a partir de vídeos por meio de visão computacional, permitindo extrair dados de vídeos já existentes.

De qualquer forma, os resultados apresentados são promissores. Em relação à tarefa de segmentação das unidades gestuais, os resultados obtidos neste trabalho são suficientes para serem avaliados por especialistas da área de estudos de gestos. Em relação à tarefa de segmentação das fases dos gestos, os resultados obtidos ao considerar o treinamento com o início do vídeo visando segmentar o restante também são interessantes e podem ser avaliados por pesquisadores da área de aplicação. A partir dessa avaliação, a construção de ferramentas capazes de auxiliar esses pesquisadores poderia ser viabilizada, fornecendo uma triagem inicial sobre os períodos de descanso e as fases dos gestos nos vídeos.

Por fim, devido ao bom desempenho para os classificadores usando Máquinas de Vetores Suporte para as classes de descanso e *stroke*, propõe-se utilizar um classificador para essas fases e uma gramática que considere as regras de produção associadas às fases dos gestos propostas por Kita et al. (1998) para identificar as demais fases – preparação, *hold* e retração.

Referências

BEGG, R. K.; PALANISWAMI, M. A machine learning approach for automated recognition of movement patterns using basic, kinetic and kinematic gait data. *Journal of Biomechanics*, v. 38, p. 401–408, 2005.

BLOOMFIELD, P. *Fourier Analysis of Time Series: An Introduction.* 2nd. ed. New York, NY, USA: John Wiley and Sons, 2000.

BRYLL, R.; QUEK, F.; ESPOSITO, A. Automatic Hand Hold Detection in Natural Conversation. In: *Proceedings of the IEEE Workshop on Cues in Communication.* Washington, DC, USA: IEEE Computer Society, 2001. p. 1–6.

CAMPBELL, W. M. Generalized linear discriminant sequence kernels for speaker recognition. In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing.* Washington, DC, USA: IEEE Computer Society, 2002. v. 1, p. 161 – 164.

CAMPS-VALLS, G.; SORIA-OLIVAS, E.; PEREZ-RUIXO, J.; PEREZ-CRUZ, F.; ARTES-RODRIGUEZ, A.; JIMENEZ-TORRES, N. Therapeutic drug monitoring of kidney transplant recipients using profiled support vector machines. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, v. 37, n. 3, p. 359 –372, may 2007.

CHEN, L.; HARPER, M.; QUEK, F. Gesture patterns during speech repairs. In: *Proceedings of 4th IEEE International Conference on Multimodal Interfaces.* Washington, DC, USA: IEEE Computer Society, 2002. p. 155–160.

CHEN, L.; LIU, Y.; HARPER, M. P.; SHRIBERG, E. Multimodal model integration for sentence unit detection. In: *Proceedings of the 6th international conference on Multimodal interfaces.* New York, NY, USA: ACM Press, 2004. p. 121–128.

CHEN, Y.; LIU, M.; LIU, J.; SHEN, Z.; PAN, W. Slideshow: Gesture-aware ppt presentation. In: *Proceedings of the 2011 IEEE International Conference on Multimedia and Expo.* Washington, DC, USA: IEEE Computer Society, 2011. p. 1 –4.

CHOI, C.; AHN, J.; BYUN, H. Visual recognition of aircraft marshalling signals using gesture phase analysis. In: *Proceedings of the Intelligent Vehicles Symposium.* Washington, DC, USA: IEEE Computer Society, 2008. p. 853 –858.

CORRADINI, A. Real-time gesture recognition by means of hybrid recognizers. In: *Gesture and sign language in human-computer interaction.* Berlin; Heidelberg; New York; Barcelona; Budapest; Hong Kong; Milan; Paris; Santa Clara; Singapure; Tokyo: Springer, 2002. p. 157–189.

CRISTIANINI, N.; SHAWE-TAYLOR, J. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge, UK: Cambridge University Press, 2000.

EISENSTEIN, J. *Gesture in Automatic Discourse Processing*. 140 p. Tese (Doutorado) — Massachussets Institute of Technology, Cambridge, MA, USA, 2008.

EISENSTEIN, J.; BARZILAY, R.; RANDALL, D. Discourse topic and gestural form. In: *Proceedings of Association for the Advancement of Artificial Intelligence*. Palo Alto, CA, USA: AAAI Press, 2008. p. 836–841.

FRASER, A. M.; SWINNEY, H. L. Independent coordinates for strange attractors from mutual information. *Physical Review A*, American Physical Society, v. 33, p. 1134–1140, 1986.

GARNHAM, A. *Psycholinguistics: Central Topics*. Londres, Reino Unido: Routledge, 1994.

GEBRE, B. G.; WITTENBURG, P.; LENKIEWICZ, P. Towards automatic gesture stroke detection. In: *Proceedings of the 8th International Conference on Language Resources and Evaluation*. Istanbul, Turkey: European Language Resources Association (ELRA), 2012.

GULLBERG, M. Some reasons for studying gesture and second language acquisition (hommage à adam kendon). *International Review of Applied Linguistics in Language Teaching*, v. 44, p. 103–124, 2006.

HAYKIN, S. *Neural networks and learning machines*. 3. ed. Upper Saddle River, NJ, USA: Prentice Hall, 2008. 936 p.

HERMANS, M.; SCHRAUWEN, B. Recurrent kernel machines: Computing with infinite echo state networks. *Neural Computation*, MIT Press, Cambridge, MA, USA, v. 24, p. 104–133, 2012.

JAAKKOLA, T. S.; HAUSSLER, D. Exploiting generative models in discriminative classifiers. In: *Proceedings of the Conference on Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 1999. p. 487–493.

KAISER, G. *A friendly guide to wavelets*. Cambridge, MA, USA: Birkhauser Boston Inc., 2011.

KANTZ, H.; SCHREIBER, T. *Nonlinear Time Series Analysis*. 2nd. ed. Cambridge, UK: Cambridge University Press, 2004.

KENDON, A. Gesticulation and speech: Two aspects of the process of utterance. In: KEY, M. R. (Ed.). *The Relationship of verbal and nonverbal communication*. The Hague, The Netherlands: Mounton Publishers, 1980. p. 207–227.

KENDON, A. An agenda for gesture studies. *The Semiotic Review of Books*, v. 7, n. 3, p. 7–12, 1996.

KENDON, A. *Gesture: Visible Action as Utterance*. Cambridge, UK: Cambridge University Press, 2005.

KETTEBEKOV, S. Exploiting prosodic structuring of coverbal gesticulation. In: *Proceedings of the 6th International Conference on Multimodal interfaces*. New York, New York, USA: ACM Press, 2004. p. 105–112.

KETTEBEKOV, S.; YEASIN, M.; KRAHNSTOEVER, N.; SHARMA, R. Prosody based co-analysis of deictic gestures and speech in weather narration broadcast. In: *Proceedings of the Workshop on Multimodal Resources and Multimodal System Evaluation*. Paris, França: European Language Resources Association, 2002.

KETTEBEKOV, S.; YEASIN, M.; SHARMA, R. Prosody based audiovisual coanalysis for coverbal gesture recognition. *IEEE Transactions on Multimedia*, v. 7, n. 2, p. 234 – 242, 2005.

KIM, D.; SONG, J.; KIM, D. Simultaneous gesture segmentation and recognition based on forward spotting accumulative HMMs. *Pattern Recognition*, v. 40, n. 11, p. 3012–3026, 2007.

KITA, S.; GIJN, I. van; HULST, H. van der. Movement phases in signs and co-speech gestures, and their transcription by human coders. In: WACHSMUTH, I.; FROHLICH, M. (Ed.). *Gesture and Sign Language in Human-Computer Interaction*. Berlin; Heidelberg; New York; Barcelona; Budapest; Hong Kong; Milan; Paris; Santa Clara; Singapure; Tokyo: Springer Berlin / Heidelberg, 1998, (Lecture Notes in Computer Science, v. 1371). p. 23–35.

KITCHENHAM, B. Procedures for performing systematic reviews. *Keele UK Keele University*, Citeseer, v. 33, n. TR/SE-0401, p. 28, 2004.

LAMAR, M.; BHUIYAN, M. S.; IWATA, A. Temporal series recognition using a new neural network structure t-combnet. In: *Proceedings of the 6th International Conference on Neural Information Processing*. New York, NY, USA: ACM Press, 1999. v. 3, p. 1112 –1117 vol.3.

LEE, L.; GRIMSON, W. Gait analysis for recognition and classification. In: *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition*. Washington, DC, USA: IEEE Computer Society, 2002. (FGR '02), p. 155–162.

LI, H.; GREENSPAN, M. Segmentation and recognition of continuous gestures. In: *Proceedings of the IEEE International Conference on Image Processing*. Washington, DC, USA: IEEE Computer Society, 2007. v. 1, p. 365 – 368.

LIMA, C. A. M.; COELHO, A. L. V. Kernel machines for epilepsy diagnosis via eeg signal classification: A comparative study. *Artificial Intelligence in Medicine*, v. 53, n. 2, p. 83–95, 2011.

LORENA, A. C.; CARVALHO, A. C. P. L. F. de. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, v. 14, n. 2, p. 43–67, 2007.

MADEO, R. C. B.; LIMA, C. A. M.; PERES, S. M. A Review on Temporal Reasoning Using Support Vector Machines. In: *Proceedings of 19th International Symposium on Temporal Representation and Reasoning*. Washington, DC, USA: IEEE Computer Society, 2012b. p. 46–52.

MADEO, R. C. B.; LIMA, C. A. M.; PERES, S. M. Gesture Unit Segmentation using Support Vector Machines: Segmenting Gestures from Rest Positions. In: *Proceedings of 28th Annual ACM Symposium on Applied Computing*. New York, NY, USA: ACM Press, 2013. p. 114–121.

MADEO, R. C. B.; PERES, S. M. *Análise Automatizada de Gestos: uma Revisão Sistemática considerando Aspectos Temporais*. São Paulo, 2012. 18 p. Disponível em: `http://www.each.usp.br/ppgsi/arquivos/RelTec/PPgSI-002_2012.pdf`.

MADEO, R. C. B.; PERES, S. M.; LIMA, C. A. M. Overview on Support Vector Machines applied to Temporal Modeling. In: *Anais do IX Encontro Nacional de Inteligência Artificial*. [S.l.: s.n.], 2012a. p. 1–6.

MARTELL, C. H.; KROLL, J. Corpus-based gesture analysis: an extension of the form dataset for the automatic detection of phases in a gesture. *International Journal of Semantic Computing*, v. 1, p. 521–536, 2007.

MCNEILL, D. *Hand and mind: What the hands reveal about thought*. Chicago, IL, USA: University of Chicago Press, 1992.

MCNEILL, D. *Gesture and Thought*. Chicago, IL, USA: University of Chicago Press, 2005.

MITCHELL, T. *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997.

NGUYEN, N. T.; BUI, T. D. Automated posture segmentation in continuous finger spelling recognition. In: *Proceedings of the 3rd International Conference on Human-Centric Computing*. Washington, DC, USA: IEEE Computer Society, 2010. p. 1 –5.

OPPENHEIM, A. V.; SCHAFER, R. W.; BUCK, J. R. et al. *Discrete-time signal processing*. 3. ed. Upper Saddle River, NJ, USA: Prentice Hall, 2009. 1120 p.

OSGOOD, B. n-dimensional Fourier Transform. In: *Lecture Notes on EE261: The Fourier Transform and its Applications*. Disponível em: `http://see.stanford.edu/materials/lsoftaee261/book-fall-07.pdf`. p. 335–402, 2007.

PHINYOMARK, A.; LIMSAKUL, C.; PHUKPATTARANONT, P. A novel feature extraction for robust emg pattern recognition. *Journal of Computing*, v. 1, p. 71–80, 2009.

QUEK, F. The Catchment Feature Model: A Device for Multimodal Fusion and a Bridge between Signal and Sense. *EURASIP Journal on Advances in Signal Processing*, v. 2004, n. 11, p. 1619–1636, 2004.

QUEK, F.; MCNEILL, D.; BRYLL, R.; DUNCAN, S.; MA, X.; KIRBAS, C.; MCCULLOUGH, K.; ANSARI, R. Multimodal human discourse: gesture and speech. *ACM Transactions on Computer-Human Interaction (TOCHI)*, ACM, v. 9, n. 3, p. 171–193, 2002.

RAMAKRISHNAN, A. S. *Segmentation of Hand Gestures using Motion Capture Data*. 67 p. Dissertação (Mestrado) — University of California, 2011.

RUDZICZ, F. Phonological features in discriminative classification of dysarthric speech. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*. Washington, DC, USA: IEEE Computer Society, 2009. p. 4605 –4608.

SCHULDT, C.; LAPTEV, I.; CAPUTO, B. Recognizing human actions: A local svm approach. In: *Proceedings of the 17th International Conference on Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2004. (ICPR '04), p. 32–36.

SEMMLOW, J. L. *Biosignal And Biomedical Image Processing: Matlab-based Applications*. New York, NY, USA: Marcel Dekker, Inc., 2004.

SHI, Z.; HAN, M. Support vector echo-state machine for chaotic time-series prediction. *IEEE Trans. on Neural Networks*, v. 18, n. 2, p. 359 –372, 2007.

SUBASI, A. EEG signal classification using wavelet feature extraction and a mixture of expert model. *Expert Systems with Applications*, n. 32, p. 1084–1093, 2007.

SUYKENS, J.; VANDEWALLE, J. Recurrent least squares support vector machines. *IEEE Transactions on Circuits and Systems-I*, v. 47, p. 1109–1114, 2000.

SWAMINATHAN, D.; THORNBURG, H.; INGALLS, T.; RAJKO, S.; JAMES, J.; CAMPANA, E.; AFANADOR, K.; LEISTIKOW, R. Capturing expressive and indicative qualities of conducting gesture: An application of temporal expectancy models. In: KRONLAND-MARTINET, R.; YSTAD, S.; JENSEN, K. (Ed.). *Computer Music Modeling and Retrieval. Sense of Sounds*. Berlin; Heidelberg; New York; Barcelona; Budapest; Hong Kong; Milan; Paris; Santa Clara; Singapure; Tokyo: Springer Berlin / Heidelberg, 2008, (Lecture Notes in Computer Science, v. 4969). p. 34–55.

VAPNIK, V. N. *Statistical Learning Theory*. New York, NY, USA: Wiley-Interscience, 1998.

VAPNIK, V. N.; CHERVONENKIS, A. Y. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probabilities and its Applications*, v. 16, p. 264–280, 1971.

VINCIARELLI, A.; PANTIC, M.; BOURLARD, H.; PENTLAND, A. Social signal processing: state-of-the-art and future perspectives of an emerging domain. In: *Proceedings of the 16th ACM International Conference on Multimedia*. New York, NY, USA: ACM, 2008. (MM '08), p. 1061–1070.

VOS, N. J. de. Echo state networks as an alternative to traditional artificial neural networks in rainfall-runoff modelling. *Hydrology and Earth Systems Sciences*, v. 17, p. 253–267, 2013.

WAN, V.; CARMICHAEL, J. Polynomial dynamic time warping kernel support vector machines for dysarthric speech recognition with sparse training data. In: *Proceedings of the Annual Conference of the International Speech Communication Association*. Red Hook, NY, USA: Curran Associates, Inc., 2005. p. 3321–3324.

WATKINS, C. Dynamic alignment kernels. MIT Press, p. 39–50, 1999.

WERAPAN, W.; CHOTIKAKAMTHORN, N. Improved dynamic gesture segmentation for thai sign language translation. In: *Proceedings of the 7th International Conference on Signal Processing*. Piscataway, NJ, USA: IEEE Standards Office, 2004. v. 2, p. 1463 – 1466 vol.2.

WILSON, A.; BOBICK, A. Realtime online adaptive gesture recognition. In: *Proceedings of the 15th International Conference on Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2000. v. 1, p. 270–275.

WILSON, A.; BOBICK, A.; CASSELL, J. Recovering the temporal structure of natural gesture. In: *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*. Washington, DC, USA: IEEE Computer Society, 1996. p. 66 –71.

WONG, S.; CIPOLLA, R. Continuous Gesture Recognition using a Sparse Bayesian Classifier. In: *Proceedings of the 18th International Conference on Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006. p. 1084–1087.

XIONG, Y.; QUEK, F. Hand Motion Gesture Frequency Properties and Multimodal Discourse Analysis. *International Journal of Computer Vision*, v. 69, n. 3, p. 353–371, 2006.

XIONG, Y.; QUEK, F.; MCNEILL, D. Hand gesture symmetric behavior detection and analysis in natural conversation. In: *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*. Washington, DC, USA: IEEE Computer Society, 2002. p. 179–184.

XIONG, Y.; QUEK, F.; MCNEILL, D. Hand Motion Gestural Oscillations and Multimodal Discourse. In: *Proceedings of the 5th International Conference on Multimodal Interfaces*. New York, NY, USA: ACM Press, 2003. p. 132–139.

YUAN, Y. *Image-based gesture recognition with support vector machines*. Tese (Doutorado) — University of Delaware, Newark, DE, USA, 2008.