Tamires Martins Rezende

Reconhecimento Automático de Sinais da Libras: Desenvolvimento da Base de Dados MINDS-Libras e Modelos de Redes Convolucionais

Belo Horizonte - Minas Gerais

Julho de 2021

Tamires Martins Rezende

Reconhecimento Automático de Sinais da Libras: Desenvolvimento da Base de Dados MINDS-Libras e Modelos de Redes Convolucionais

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito parcial para obtenção do Título de Doutora em Engenharia Elétrica.

Universidade Federal de Minas Gerais - UFMG

Programa de Pós-Graduação em Engenharia Elétrica - PPGEE

Machine Intelligence and Data Science Laboratory - MINDS

Orientador: Frederico Gadelha Guimarães

Coorientadora: Sílvia Grasiella Moreira Almeida

Belo Horizonte - Minas Gerais

Julho de 2021

Rezende, Tamires Martins.
R467r Reconhecimento automático de sinais da Libras [recurso eletrônico] : desenvolvimento da base de dados MINDS-Libras e modelos de redes convolucionais / Tamires Martins Rezende. - 2021.
1 recurso online (179 f. : il., color.) : pdf.

Orientador: Frederico Gadelha Guimarães.
Coorientadora: Sílvia Grasiella Moreira Almeida.

Tese (doutorado) - Universidade Federal de Minas Gerais, Escola de Engenharia.

Apêndices: f. 143-179.

Bibliografia: f. 121-141.
Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia elétrica - Teses. 2. Aprendizado profundo - Teses. 3. Língua brasileira de sinais - Teses. 4. Língua de sinais - Teses. 5. Redes neurais convolucionais - Teses. I. Guimarães, Frederico Gadelha. II. Almeida, Sílvia Grasiella Moreira. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.

CDU: 621.3(043)

Ficha catalográfica elaborada pelo bibliotecário Reginaldo Cesar Vital dos Santos - CRB-6/2165
Biblioteca Prof. Mário Werneck, Escola de Engenharia da UFMG.

"Reconhecimento Automático de Sinais da Libras: Desenvolvimento da Base de Dados MINDS-Libras e Modelos de Redes Convolucionais"

Tamires Martins Rezende

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Doutor em Engenharia Elétrica.

Aprovada em 23 de julho de 2021.

Por:

Prof. Dr. Frederico Gadelha Guimarães
DEE (UFMG) - Orientador

Prof. Dr. Sílvia Grasiella Moreira Almeida
(IFMG Ouro Preto)

Prof. Dr. Cleber Zanchettin
Centro de Informática (UFPE)

Prof. Dr. David Menotti Gomes
DINF (UFPR)

Prof. Dr. Silvia Silva da Costa Botelho
Centro de Ciências Computacionais (FURG)

Prof. Dr. Hani Camille Yehia
DELT (UFMG)

À minha família e à pequena Tamires Martins Rezende que sempre teve o sonho de ensinar.

Agradecimentos

Gratidão... do fundo do meu coração!

Sensação de dever cumprido... de dedicação e muito esforço. Este doutorado simboliza a realização de uma grande conquista pessoal.

Agradeço a Deus por iluminar os meus caminhos e me proteger. Agradeço a Ele por colocar pessoas tão boas no meu caminho, permitindo que eu nunca me sentisse desamparada. Não foi fácil, mas sempre senti a sua presença!

À minha família por ser a base da minha vida: minha mãe Nelita pelos sucos e frutinhas; meu pai Emidio pelas orações; aos meus irmãos Ronaldo, Luciano e Viviane, minhas inspirações; aos meus cunhados Cristina e Emerson pela solicitude; aos meus sobrinhos, meus pacotes de amor: Ronaldo Júnio, Heitor, Manuela e Nicolas, razões de todos os meus atrasos. Definitivamente, tia/dinda não consegue viver sem vocês. Muito obrigada pela torcida. Amo vocês! Estendo os meus agradecimentos a todos tios/tias/primos/primas que sempre estiveram presentes com palavras de motivação e carinho.

À Sílvia Almeida, uma mulher inspiradora e uma orientadora maravilhosa. Eu não tenho palavras pra descrever o quanto eu te admiro. Me sinto lisonjeada por ter continuado o seu trabalho e por ter ganhado uma amiga. O carinho que você sempre teve com a minha pesquisa e com a minha pessoa é indescritível. Você é um exemplo.

Ao Frederico Gadelha, um super orientador... muito mais do que uma IA. Obrigada pelos desafios que me propôs, por entender as minhas limitações e por me motivar constantemente. Eu não consigo descrever o quanto a sua orientação foi importante para mim. Obrigada pelo carinho e por tornar tudo mais leve.

Ao Cristiano Leite, por ter me ensinado o que é pesquisa e o que é ser pesquisadora. Obrigada por ter me acolhido com tanto carinho na UFMG.

Ao time Minds pela parceria, discussões, *shots*, cachaças, enfim... tudo com vocês é enriquecedor. Eu desconheço um laboratório tão dinâmico e acolhedor. Vocês me ensinaram mais do que pesquisa... aprendi com vocês que eu sou capaz. Um agradecimento especial ao Marcos Alves pela presença, por ser suporte, por ser meu grande amigo; ao Antônio Carlos e Patrícia de Oliveira, por tanta atenção e ajuda, trabalhar com vocês é sensacional; e à Rúbia Reis, Moises Mendes e Giulia Zanon, por terem participado desta pesquisa e me

ensinado o caminho das pedras.

Aos meus amigos mais que especiais: Gabi e Robinho, por me apoiarem e deixarem a pandemia mais leve. Saber que eu posso contar com vocês deixa os meus dias mais alegres. À Letícia Resende, Jaime Arturo e Aline Fidêncio por dividirem comigo essa jornada. Sabemos que não é fácil, mas que conseguimos! À Maria Astélia, Izabela Neves, Aline Palheiros, Hortência Franco e Bruna Silveira, por deixarem os dias em BH mais agradáveis.

Ao IFMG, em especial ao *Campus* Avançado Itabirito. Aos amigos: Elias Rezende, Luiz Olmes, Kleber Mazione e Marcuuuus Diadelmo. Obrigada pelo apoio que sempre me deram e por alegrarem o meu caminho. Vocês são especiais.

À Gaia, especialmente Adriano Lisboa e Pedro Venâncio. Obrigada por me acolherem e me proporcionarem tanto aprendizado.

A todos os professores que tive na minha jornada acadêmica, que me mostraram como a educação é essencial e que ela é sim a solução de todos os problemas da sociedade. A educação muda os caminhos e abre a mente. Meu muito obrigada às instituições por onde eu passei.

À Silvia Botelho, Cléber Zanchettin, David Menotti e Hani Yehia por avaliarem este trabalho e ao Flávio Cardeal pelas contribuições realizadas na qualificação.

Por fim, agradeço ao programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais e ao CNPq pelo apoio financeiro durante a realização deste trabalho.

Gratidão a todos!

Obrigada, obrigada e obrigada!

"A inteligência artificial está na raiz da mudança de época que estamos vivendo. A robótica pode tornar possível um mundo melhor se estiver unida ao bem comum. Porque se o progresso tecnológico aumenta as desigualdades, não é um progresso real. Os avanços futuros devem estar orientados para o respeito pela dignidade da pessoa e da Criação. Rezemos para que o progresso da robótica e da inteligência artificial esteja sempre a serviço do ser humano... podemos dizer, que 'seja humano'."

Papa Francisco

Resumo

O reconhecimento automático da Língua de Sinais tem sido um desafio para a área de Inteligência Computacional, dada a natureza visual-gestual que configura esse complexo sistema de comunicação. Esta tese se insere nesse contexto e foca os esforços na língua brasileira de sinais, Libras. Para isso, uma nova base de dados chamada MINDS-Libras foi proposta. Ela contém (i) vídeos em RGB, (ii) vídeos com informação de profundidade, (iii) informações de 25 pontos/juntas do corpo e de (iv) 1347 pontos da face do sinalizador. Cada um dos 20 sinais que compõem essa base foi gravado 5 vezes por 12 sinalizadores, totalizando 1200 amostras. Utilizando esses dados, duas diferentes arquiteturas de Aprendizado Profundo foram propostas para reconhecimento dos sinais da MINDS-Libras. A primeira delas foi uma Rede Neural Convolucional 3D utilizando vídeos e, a segunda, uma Rede Neural Convolucional Temporal para a trajetória manual. A abordagem que apresentou menor sensibilidade à mudança do sinalizador foi aquela cujo dado de entrada era o movimento das mãos, podendo esse ser considerado o parâmetro mais importante para a formação do sinal. Os resultados também indicam que esse tipo de abordagem é viável para o reconhecimento dos sinais da Libras. Novas perspectivas podem ser abertas com a expansão da base de dados e inclusão de mais sinalizadores no processo de gravação de (novos) sinais.

Palavras-chave: Aprendizado Profundo, Redes Neurais Convolucionais, Reconhecimento Automático da Libras, Língua de Sinais, Libras.

Abstract

The automatic recognition of Sign Language has been a challenge for the Computational Intelligence area, given the visual-gestural nature that configures this complex communication system. This thesis falls within this context and focuses efforts on the Brazilian Sign Language, Libras. For this purpose, a new database called MINDS-Libras has been proposed. It contains (i) RGB videos, (ii) videos with depth information, (iii) information from 25 *points/joints of the body and from (iv)* 1347 *points of the face of the signaller. Each of the* 20 *signs that build this base were recorded* 5 *times by* 12 *signallers, totaling* 1200 *samples. Using this data, two different Deep Learning architectures were proposed for recognizing the MINDS-Libras signs. The first one was a 3D Convolutional Neural Network by using videos, and the second a Temporal Convolutional Neural Network for the manual trajectory. The best leave-one-signaller-out was that based in the hand movement, and this can be considered the most important parameter for sign formation. The results also indicate that this approach is feasible for the Libras signs recognition. New perspectives may be opened with the expansion of the database and add more signallers in the process of recording (new) signs.*

Keywords: Deep Learning, Convolutional Neural Network, Sign Language Recognition, Sign Language, Libras.

Lista de Figuras

Figura 1 – Metodologia proposta para o Reconhecimento Automático de Sinais da Libras. 22

Figura 2 – Parâmetros fonológicos da Libras: Ponto de articulação (PA), Configuração de mão (CM), Movimento (M), Orientação da palma da mão (Or) e Expressões não-manuais (ENM). 28

Figura 3 – Distribuição das línguas de sinais no mundo. 29

Figura 4 – Etapas básicas de um sistema de reconhecimento de gestos. 30

Figura 5 – Sistema de aquisição dos sinais acoplado na cabeça. 31

Figura 6 – Sistema de aquisição dos sinais utilizando um celular. 31

Figura 7 – Histórico de publicações que realizaram o reconhecimento automático de sinais (Dados coletados até maio/2021). 33

Figura 8 – Segmentação de sinal por meio do movimento labial: (a) detecção de movimento, (b) detecção das mãos e lábios, (c) exemplo de extração de lábio, e (d) segmentação de uma frase composta por dois sinais, analisando a área dos lábios *versus* os quadros na sequência de um vídeo. 34

Figura 9 – Número de publicações que utilizaram técnicas de extração de características baseadas em Visão Computacional ou em Sensores acoplados ao indivíduo (Dados coletados até maio/2021). 35

Figura 10 – Coleta de dados utilizando técnicas de Visão Computacional e Sensores acoplado às mãos. 36

Figura 11 – Número de publicações que aplicaram o Aprendizado Profundo nas pesquisas de Visão Computacional (Dados coletados até maio/2021). . 37

Figura 12 – Número de publicações relacionadas à trabalhos de reconhecimento de sinais em cada língua de sinais (Dados coletados até maio/2021). . . . 39

Figura 13 – Exemplo de um quadro e os 100 pontos faciais extraídos de um quadro da base *Grammatical Facial Expressions Data Set*. 45

Figura 14 – Quadros de uma sequência de vídeo da base de dados Libras-34. 45

Figura 15 – 12 quadros RGB de uma amostra da base de dados Libras-10. 46

Figura 16 – Exemplo de sinal da base de dados LIBRAS-Ufop. 46

Figura 17 – Posição de descanso definida para a gravação de cada amostra. 49

Figura 18 – Sinais que compõem o banco de dados MINDS-Libras. 50
Figura 19 – Visualização dos sinais da MINDS-Libras utilizando a técnica t-SNE. . 51
Figura 20 – Dados disponibilizados pelos dispositivos de captura: câmera RGB e sensor RGB-D. 53
Figura 21 – Cenário criado para a gravação dos sinais. Ambiente interno, com iluminação controlada. 55
Figura 22 – Organização dos dados disponibilizados referentes à câmera RGB. . . . 56
Figura 23 – Organização dos dados disponibilizados referente ao sensor RGB-D. . . 57
Figura 24 – Posição das 25 juntas capturadas pelo sensor RGB-D: 1−base da coluna, 2−coluna, 3−pescoço, 4−cabeça, 5-ombro esquerdo, 6-cotovelo esquerdo, 7-pulso esquerdo, 8-mão esquerda, 9-ombro direito, 10-cotovelo direito, 11-pulso direito, 12-mão direita, 13-quadril à esquerda, 14-joelho esquerdo, 15-tornozelo esquerdo, 16-pé esquerdo, 17-quadril à direita, 18-joelho direito, 19-tornozelo direito, 20-pé direito, 21-ombro/coluna, 22-ponta da mão esquerda, 23-polegar esquerdo, 24-ponta da mão direita e 25-polegar direito. 62
Figura 25 – Referência dos eixos com base ao centro geométrico do sensor RGB-D. 63
Figura 26 – Posição dos pontos do corpo para um quadro de uma amostra para x ∈ $[-2.2, 2.2]$, y ∈ $[-1.6, 1.6]$ e z ∈ $[0, 4]$. 63
Figura 27 – *TrackingState* de um quadro. 64
Figura 28 – Estados de cada uma das mãos ao londo dos quadros de uma amostra. 64
Figura 29 – *HandState* das mãos referente ao quadro 30 apresentado na Figura 28. 65
Figura 30 – 17 pontos do corpo plotados no quadro em RGB. 65
Figura 31 – Posição x-y dos 17 pontos do corpo plotados no quadro em profundidade. 66
Figura 32 – Variação das coordenadas x (verde '-.'), y (vermelho '-') e z (azul '.') normalizadas para cada um dos 17 pontos do corpo, ao longo dos 150 quadros da amostra *1-01Acontecer_1Body.txt*. 67
Figura 33 – Variação das coordenadas x ('-.'), y ('–') e z ('-') normalizadas do ponto 1 (base da coluna) para os 20 sinais da MINDS-Libras (sinalizador 1, gravação 1). 68
Figura 34 – Variação das coordenadas x ('-.'), y ('–') e z ('-') normalizadas do ponto 1 (base da coluna) para cada um dos sinalizadores da MINDS-Libras (Sinal "acontecer", gravação 1). 68
Figura 35 – Variação das coordenadas x, y e z normalizadas dos 5 pontos relativos à mão esquerda, ao longo dos 150 quadros da amostra *1-01Acontecer_1Body.txt*. 69
Figura 36 – Variação das coordenadas x, y e z normalizadas dos 5 pontos relativos à mão direita, ao longo dos 150 quadros da amostra *1-01Acontecer_1Body.txt*. 69

Figura 37 – Variação das coordenadas x (verde '-.'), y (vermelho '-') e z (azul '–') normalizadas do ponto 22-PontaMaoEsquerda para as cinco gravações do sinal "acontecer", sinalizador 01. 70
Figura 38 – Variação das coordenadas x normalizadas do ponto 22-PontaMaoEsquerda para a gravação 1 de cada sinalizador (Sinal "acontecer"). 70
Figura 39 – *FaceBox*: coordenadas do retângulo que delimita a face do sinalizador, plotado no quadro RGB correspondente (Sinalizador 4). 71
Figura 40 – Exemplos e orientação da cabeça. 72
Figura 41 – *FaceModel*: coordenadas dos 1347 pontos relativos à face do sinalizador e *HeadPivot*: ponto de referência para o movimento da cabeça destacado em vermelho (*) (Sinalizador 4). 72
Figura 42 – *ColorFaceModel*: posição x-y dos 1347 pontos do corpo plotados no quadro em RGB (Sinalizador 4). 73
Figura 43 – *DepthFaceModel*: posição x-y dos 1347 pontos do corpo plotados no quadro em profundidade (Sinalizador 4). 73
Figura 44 – *FaceModel* e os 37 pontos (preto '*') representativos da face. 74
Figura 45 – Velocidade média das mãos na amostra *2-07Banco_ 5RGB.mp4* (Sinalizador: 2, Sinal: Banco, Gravação: 5). Limiar para corte nos quadros 13 e 79. 75

Figura 46 – Etapas básicas de um processo de aprendizado de máquina. 77
Figura 47 – Quadros iniciais (1 a 12) da amostra *6-01Acontecer_ 2RGB.mp4* (Sinalizador: 6, Sinal: Acontecer, Gravação: 2) que não caracterizam movimento. 79
Figura 48 – Quadros finais (80 a 150, de 5 em 5) da amostra *6-01Acontecer_ 2RGB.mp4* (Sinalizador: 6, Sinal: Acontecer, Gravação: 2) que não caracterizam movimento. 79
Figura 49 – Quadros 22, 38, 48, 67 e 79 retornados da sumarização aplicada na amostra *2-07Banco_ 5RGB.mp4* (Sinalizador: 2, Sinal: Banco, Gravação: 5). 80
Figura 50 – Quadros com movimento (13 ao 79) da amostra *2-07Banco_ 5RGB.mp4* (Sinalizador: 2, Sinal: Banco, Gravação: 5) e os 5 quadros retornados da sumarização, em destaque: 22, 38, 48, 67 e 79. 81
Figura 51 – Arquitetura da CNN 3D proposta. 83
Figura 52 – Treinamento do modelo CNN3D. 85
Figura 53 – Matriz de entrada com 30 séries temporais (coordenadas x-y-z) em 150 quadros para cada uma das 1200 amostras. Em destaque: equação referente à terceira série temporal (coordenada x, $p = 8$) do segundo quadro ($q = 2$) na primeira amostra. 87
Figura 54 – *Data augmentation* relativo à série temporal do ponto 8-MãoEsquerda (coordenada Y) da amostra *1-01Acontecer_ 1Body.txt*. 88

Figura 55 – Arquitetura TCN utilizada. 89
Figura 56 – Treinamento do modelo TCN. 90

Figura 57 – Etapas das abordagens propostas nesta tese para o Reconhecimento Automático de Sinais da Libras. 92
Figura 58 – Matriz de confusão normalizada obtida pela média 12-*folds*. 94
Figura 59 – Curva ROC da CNN 3D (AUC $= 0,92$). 94
Figura 60 – Sinais que tiveram as maiores taxas de reconhecimento. 96
Figura 61 – Sinal "banheiro" classificado erroneamente como sinal "acontecer" e sinal "esquina" classificado erroneamente como sinal "banheiro". 97
Figura 62 – Sinal "aproveitar" que apresentou pior desempenho nas métricas analisadas. 97
Figura 63 – Saída do primeiro bloco convolucional da CNN 3D proposta (*Conv3D_1*) na execução do sinal "banco": 3 imagens 222×222 para cada um dos 4 mapas de características. 98
Figura 64 – Matriz de confusão normalizada obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinal. 100
Figura 65 – Curva ROC da TCN quando os conjuntos de treino e teste foram divididos por sinal (AUC $= 0,98$). 100
Figura 66 – Matriz de confusão normalizada obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinalizador. . 102
Figura 67 – Sinal "vacina" classificado erroneamente como sinal "aluno". 102
Figura 68 – Sinal "esquina" classificado erroneamente como sinal "sapo". 103
Figura 69 – Curva ROC da TCN quando os conjuntos de treino e teste foram divididos por sinalizador. 104

Figura 70 – Alfabeto manual da Libras. 133
Figura 71 – Configurações de mão da Libras. 133
Figura 72 – Exemplos de ponto de articulação em Libras: sinal (a) "baleia" e (b) "escova de dente". 134
Figura 73 – Exemplos de movimento em Libras: sinal (a) "cadeira" e (b) "sentar-se". 135
Figura 74 – Orientação da palma da mão em Libras. 136
Figura 75 – Exemplos de orientação da mão em Libras: sinal (a) "ajudar alguém" e (b) "ser ajudado". 137
Figura 76 – Expressão facial negativa do sinal "não sei". 137

Lista de Tabelas

Tabela 1 – Trabalhos que aplicaram técnicas de Aprendizado Profundo. 38

Tabela 2 – Bases de Línguas de Sinais publicadas. 44
Tabela 3 – Características desejáveis para uma base de dados da língua de sinais. . 48
Tabela 4 – Características dos sinalizadores. 52
Tabela 5 – Arquivo do sensor RGB-D relativo aos dados do corpo. 56
Tabela 6 – Arquivo do sensor RGB-D relativo aos dados da face. 57
Tabela 7 – Número de quadros referente aos vídeos da câmera RGB para os sinais: (01) acontecer, (02) aluno, (03) amarelo, (04) América, (05) aproveitar, (06) bala, (07) banco, (08) banheiro, (09) barulho, (10) cinco, (11) conhecer, (12) espelho, (13) esquina, (14) filho, (15) maçã, (16) medo, (17) ruim, (18) sapo, (19) vacina e (20) vontade. 59

Tabela 8 – Parâmetros da arquitetura. 83
Tabela 9 – Determinação dos parâmetros da TCN. 91
Tabela 10 – Características dos experimentos com a base de dados MINDS-Libras . 91

Tabela 11 – Métricas de desempenho da abordagem com a CNN 3D. 96
Tabela 12 – Análise dos pontos das mãos. 99
Tabela 13 – Métricas de desempenho obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinal. 101
Tabela 14 – Desempenho médio de cada sinalizador nas 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinalizador. 104
Tabela 15 – Métricas de desempenho obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinalizador. 105
Tabela 16 – Resultados dos experimentos realizados com a base de dados MINDS-Libras 105

Tabela 17 – Descrição dos sinais gravados. Notação: Mão Direita (MD), Mão Esquerda (ME). 140

Lista de Abreviaturas, Siglas e Símbolos

#	Número, quantidade.
≈	Aproximadamente.
3D	Três dimensões.
ASL	Língua Americana de Sinais (*American Sign Language*).
CBA	Congresso Brasileiro de Automática.
CM	Configuração de Mão.
CNN	Rede Neural Convolucional (*Convolutional Neural Network*).
CNN 3D	Rede Neural Convolucional 3D.
Conv3D	Camada Convolucional 3D.
DBN	*Dynamic Bayesian Network*.
ELM	*Extreme Learning Machine*.
ENIAC	Encontro Nacional de Inteligência Artificial e Computacional.
ENM	Expressões Não-Manuais.
FC	Totalmente Conectada (*Fully-Conected*).
FN	Falso Negativo.
FP	Falso Positivo.
FSL	Língua Francesa de Sinais (*French Sign Language*).
GPU	Unidade de Processamento Gráfico (*Graphics Processing Unit*).
HMM	Modelos Ocultos de Markov.

IA	Inteligência Artificial.
ISL	Língua Indiana de Sinais (*Indian Sign Language*).
JSL	Língua Japonesa de Sinais (*Japan Sign Language*).
L1	Língua materna, primeira língua.
L2	Segunda língua.
LA-CCI	*Latin American Conference on Computational Intelligence.*
Libras	Língua Brasileira de Sinais.
LSTM	*Long short-term memory.*
M	Movimento.
MaxPool3D	Camada *Max Pooling* 3D.
MD	Mão Direita.
ME	Mão Esquerda.
Minds	*Machine Learning and Data Science.*
MSES	*Memetic Self-Adaptive Evolution Strategies.*
Or	Orientação da Mão.
PA	Ponto de Articulação ou Locação.
PDM	Problema da Diversidade Máxima.
ReLU	*Rectified Linear Unity.*
RGB	Vermelho, Verde e Azul (*Red, Green and Blue*).
RGB-D	Vermelho, Verde, Azul e Profundidade (*Red, Green, Blue and Depth*).
RNA	Rede Neural Artificial.
RNN	Rede Neural Recorrente (*Recurrent Neural Networks*).
ROI	Região de Interesse (*Region Of Interest*).
SIBI	*Sistem Isyart Bahasd Indonesia.*
SLR	Reconhecimento da Língua de Sinais (*Sign Language Recognition*).
SBAI	Simpósio Brasileiro de Automação Inteligente.

SVM	Máquina de Vetores de Suporte (*Support Vector Machine*).
t–SNE	*T–distributed Stochastic Neighbor Embedding.*
TCN	Rede Neural Convolucional Temporal.
UFMG	Universidade Federal de Minas Gerais.
UFOP	Universidade Federal de Ouro Preto.
UCI	*University of California Irvine.*
VP	Verdadeiro Positivo.

Sumário

1 Introdução 20
1.1 Motivação e Caracterização do Problema 21
1.2 Objetivos 23
1.3 Principais Contribuições da Tese 24
1.4 Estrutura do Documento 24
2 Revisão Bibliográfica 26
2.1 Breve Contexto da Língua Brasileira de Sinais 26
2.2 Reconhecimento Automático da Língua de Sinais 29
2.3 Principais Contribuições do Capítulo 40
3 MINDS-Libras Dataset 42
3.1 Bases de Dados disponíveis na Literatura 43
3.2 Protocolo de Gravação 47
3.2.1 Sinais Escolhidos 49
3.2.2 Sinalizadores 52
3.2.3 Sensores 53
3.2.4 Cenário das Gravações 54
3.2.5 Dados Disponibilizados 54
3.3 Análise Descritiva 57
3.3.1 Dados da câmera RGB: vídeos 58
3.3.2 Dados do sensor RGB-D: juntas corpo 62
3.3.3 Dados do sensor RGB-D: pontos da face 71
3.3.4 Dados do sensor RGB-D: vídeos 74
3.4 Principais Contribuições do Capítulo 75
4 Reconhecimento de Sinais da Libras 77
4.1 Abordagem utilizando vídeos gravados em padrão RGB 78
4.1.1 Pré-processamento 78
4.1.2 Escolha do modelo: arquitetura CNN3D 82
4.1.3 Treinamento 84
4.1.4 Análise do modelo 85
4.2 Abordagem utilizando a informação das mãos 86
4.2.1 Pré-processamento 86
4.2.2 Escolha do modelo: arquitetura TCN 88
4.2.3 Treinamento 89
4.2.4 Análise do modelo 90
4.3 Principais Contribuições do Capítulo 91
5 Resultados 92

5.1 Abordagem utilizando vídeos gravados em padrão RGB 93
5.2 Abordagem utilizando a informação das mãos 99
5.3 Principais Contribuições do Capítulo 104
6 Conclusões 106
6.1 Abordagens Propostas 107
6.2 Futuras Investigações 109
6.3 Publicações 110

Referências 111

Apêndice A Parâmetros Fonológicos da Língua de Sinais 132
A.1 Configuração de Mão 132
A.2 Ponto de Articulação ou Locação 134
A.3 Movimento 134
A.4 Orientação da Mão 136
A.5 Expressões Não-Manuais 136
Apêndice B Descrição dos Sinais Gravados 138
Apêndice C Publicações 143
C.1 Facial Expression Analysis in Brazilian Sign Language for Sign Language. . 143
C.2 Desenvolvimento de uma Base de Dados de Sinais de Libras para Aprendizado de Máquina: Estudo de Caso com CNN 3D. 150
C.3 Development and Validation of a Brazilian Sign Language Database for Human Gesture Recognition. 154
C.4 Trabalhos complementares. 165

Capítulo 1

Introdução

A transmissão de informação é a base da comunicação. É por meio dos sentidos sensoriais que as pessoas demonstram a capacidade de expressar ideias e, com base nas estruturas linguísticas, o diálogo entre duas partes é estabelecido. Essa troca de informação pode ocorrer de diferentes formas, sendo a verbal (oral-auditiva) e a por meio de sinais (visual-gestual) as mais comumente utilizadas. Dentro da segunda categoria encontram-se as línguas de sinais, que utilizam estímulos caracterizados por movimentos manuais, corporais e expressões faciais, estabelecendo a comunicação com e entre pessoas privadas de audição, seja ela total ou parcial.

No Brasil, a língua de sinais é conhecida por Libras (**LÍ**ngua **BRA**sileira de **S**inais) e é por meio dela que a comunidade surda se comunica. É uma língua reconhecida oficialmente em 2002 pela Lei n^{o} 10.436 (Brasil, 2002) e, como todo idioma, está constantemente em processo de evolução, principalmente quando se trata de vocabulário. Em 2005 o Decreto n^{o} 5.626 (Brasil, 2005) possibilitou a regulamentação da Libras, permitindo que temas como a inclusão e a acessibilidade dos surdos na sociedade fossem mais discutidos. Desde então existem leis que tornam obrigatório o ensino da Libras em cursos de licenciatura; que exigem a disponibilidade de um intérprete em ambientes públicos para possibilitar o atendimento de um surdo; e que impõem a presença de tradutores ou intérpretes da Libras em eventos em que surdos estejam presentes.

Apesar da legislação que ampara a língua, a comunidade surda ainda se depara com barreiras de comunicação constantemente. Os ouvintes, em sua maioria, se encontram despreparados para se comunicar em Libras. Alguns não conhecem a língua e uma outra parcela acredita que existe uma língua de sinais universal. Esse desconhecimento não acontece apenas para os usuários da língua oral, há muitos surdos que também não são apresentados a ela. A falta de disseminação da Libras está presente na sociedade em geral, principalmente nos grupos que defendem o oralismo na educação dos surdos (Rodrigues, 2008), impossibilitando que os mesmos aprendam esta forma de comunicação completa e estruturada.

Diante desse cenário, este estudo apresenta uma metodologia que permita o reconhecimento automático de sinais[1] da Libras, como um suporte na comunicação surdo-ouvinte. Baseando-se em técnicas de Aprendizado de Máquina (*Machine Learning*), este trabalho detalha: (i) a aquisição dos dados, (ii) a criação da base de dados MINDS-Libras, (iii) sua análise descritiva, (iv) as técnicas de pré-processamento utilizadas e (v) a classificação dos sinais. Toda essa estrutura foi pensada, inicialmente, no ouvinte, pois busca realizar a tradução de sinais em Libras para a palavra em Língua Portuguesa. Entretanto o surdo é um beneficiário, consequentemente, devido ao fato da metodologia ter sido elaborada para diminuir a barreira de comunicação entre os surdos e não surdos, estimulando o aprendizado da língua brasileira de sinais.

Para contextualizar esta pesquisa, este capítulo apresenta na Seção 1.1 o problema abordado, expondo a hipótese de trabalho. Já nas Seções 1.2 e 1.3 foram descritos os objetivos e as principais contribuições desta tese, respectivamente. Por fim, na Seção 1.4 encontra-se a forma como este texto foi organizado.

1.1 Motivação e Caracterização do Problema

A estrutura metodológica desenvolvida para reconhecimento automático de sinais da Libras é ilustrada pela Figura 1. O escopo do projeto foi definido após uma revisão bibliográfica que trouxe as principais características que um estudo com tal objetivo deveria conter. O reconhecimento de sinais é um tópico muito estudado na literatura, proveniente da área de reconhecimento de gestos que, por sua vez, engloba quaisquer tarefas relacionadas com a classificação e detecção de movimentos corporais.

Mesmo com sua importância social, há poucos estudos que se propõem a realizar o seu reconhecimento automático no caso da língua brasileira. De acordo com Athira et al. (2019), essa lacuna na pesquisa começa com a falta de uma base de dados que permita a validação de metodologias de classificação. Validar uma metodologia é uma tarefa que poderia ser realizada com quaisquer conjuntos de dados, independente do idioma, mas o esforço desta pesquisa para estruturar uma base de dados focada na Libras tem como objetivo proporcionar evolução e visibilidade para a área no Brasil.

Na literatura, há duas abordagens principais quando se fala de aquisição de dados: (i) as que utilizam sensores "vestíveis" (Li et al., 2015b; Tubaiz et al., 2015; Gałka et al., 2016; Júnior et al., 2017; Kakoty e Sharma, 2018; Gupta e Kumar, 2021), e (ii) as que utilizam imagens/vídeos e aplicam Visão Computacional (Kim et al., 2017; Kumar et al., 2018e; Li et al., 2018; Koller et al., 2018; Lim et al., 2019; Ferreira et al., 2019; Sharma et al., 2020; Lee et al., 2021). A primeira é uma técnica invasiva e relativamente cara. Já a

[1] O sinal é a menor unidade da língua de sinais. Ele equivale às palavras nas línguas orais (Quadros e Karnopp, 2004), e é formado pela combinação de parâmetros visuais, como movimento, orientação da mão, expressão facial, ponto de articulação e configuração da mão.

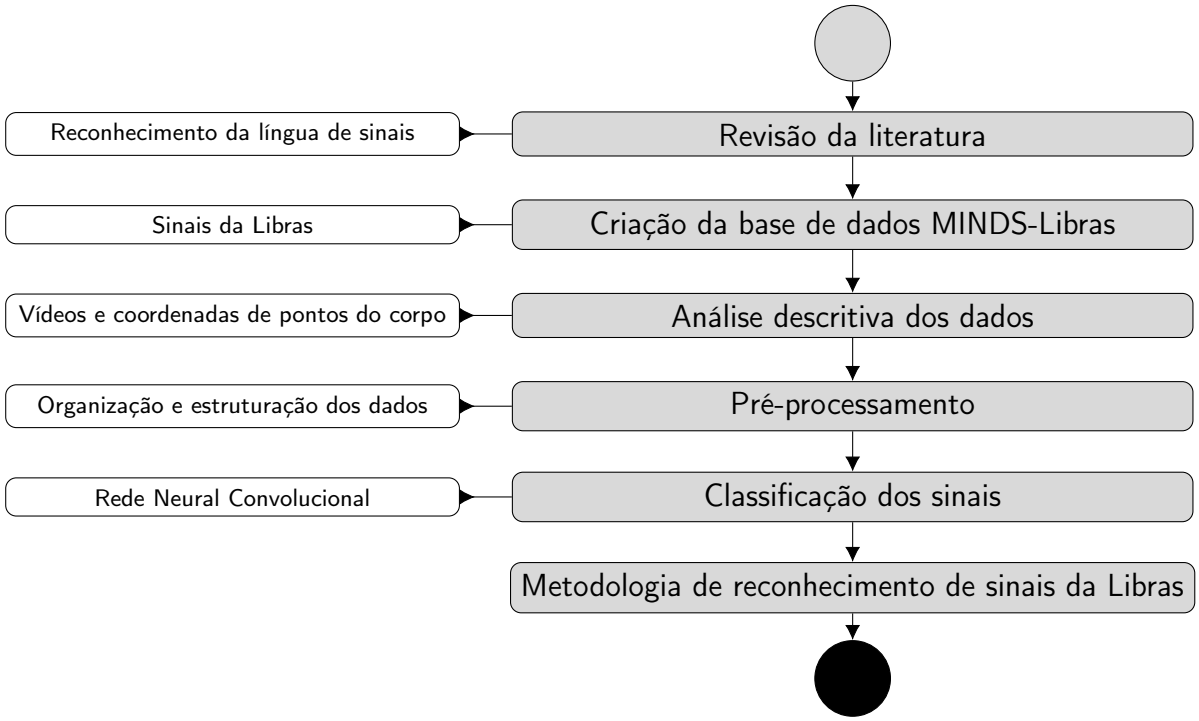


Figura 1 – Metodologia proposta para o Reconhecimento Automático de Sinais da Libras.

segunda requer um processamento custoso, mas pertence a uma área em que as tecnologias suprem essa desvantagem.

Baseada em Visão Computacional, a MINDS-Libras foi criada por meio da aquisição dos dados com uma câmera comercial e um sensor RGB-D (*Red, Green, Blue and Depth*). Os resultados foram (i) vídeos em formato RGB (*Red, Green, Blue*), (ii) vídeos em profundidade, (iii) informação espacial de 25 juntas do corpo e (iv) de 1347 pontos da face para cada execução dos sinais. A MINDS-Libras contém 20 sinais gravados 5 vezes por 12 sinalizadores[2]. Isso significa que 1200 amostras para cada tipo de dado foram disponibilizadas publicamente, mapeando expressão facial, movimento dos braços, mãos e corpo. A escolha dos sinais que compõem a base foi realizada por um especialista da língua, tendo como critério a diversificação em relação às unidades formacionais de um sinal. O tamanho da base motivou a realização de um estudo exploratório para que fosse possível investigar as peculiaridades de cada informação. Essa análise foi fundamental, pois foi por meio dela que se realizou uma documentação mais detalhada dos dados, possibilitando a visualização de padrões não notados anteriormente.

Como a base MINDS-Libras possui dados em vídeo e o posicionamento espacial de vários pontos do corpo, técnicas de classificação que podem ser aplicadas a dados vindos de sensores "vestíveis" ou de imagens/vídeos tornaram-se promissoras para este trabalho. Mesclar as abordagens mostra a possibilidade de intercâmbio entre as técnicas e amplia as possibilidades de atuação da metodologia desenvolvida neste trabalho.

[2] Sinalizador é a pessoa que executa o sinal.

Dadas as várias opções presentes na literatura, a abordagem descrita nesta tese explorou uma estrutura que fosse menos sensível ao sinalizador, uma vez que cada pessoa tem a sua forma específica de executar o sinal. Isso indica que, como na língua oral, a forma de transmitir as informações apresenta variações, como gírias e sotaques, oriundos de cada individuo. Serrão et al. (2021) expõe essa característica da língua de sinais como um desafio para os algoritmos de reconhecimento de padrões.

Nesse contexto, partiu-se da hipótese que, tendo uma base de dados de sinais da Libras, a metodologia mais adequada para reconhecer automaticamente os sinais, independente do sinalizador (*leave-one-signaller-out*), é considerando a informação temporal do movimento e a trajetória das mãos. Para isso, optou-se pelas Redes Neurais Convolucionais (CNN) para classificar os sinais em duas perspectivas: (i) a CNN 3D quando os dados de entrada forem os vídeos e (ii) a rede neural convolucional temporal (TCN) quando forem utilizados os pontos do corpo. Parte-se do pressuposto que a segunda perspectiva seja a abordagem de menor sensibilidade ao sinalizador e, consequentemente, a menos custosa computacionalmente. Além da abordagem *leave-one-signaller-out*, as duas metodologias foram aplicadas no cenário em que a dependência do sinalizador fosse considerada, isto é, na divisão dos dados de treino e teste por sinal.

Em suma, este trabalho utiliza dados reais em um aspecto social que foca na classificação de padrões da língua. Muitos foram os desafios encontrados por este se tratar de um problema real que possui diversas ramificações, por ter como material uma língua pouco explorada na comunidade científica e, também, por limitações de implementação e hardware. Entretanto, abordagens para o reconhecimentos de sinais da Libras foram estruturadas buscando explorar a área, a língua e descobrindo soluções para as dificuldades encontradas.

1.2 Objetivos

O objetivo desta tese é desenvolver uma metodologia que realize o reconhecimento automático de sinais da Libras, focando numa classificação que possibilite uma menor sensibilidade do modelo ao sinalizador. Para que este propósito seja alcançado, os seguintes objetivos específicos foram estabelecidos:

1. Realizar um levantamento do estado da arte no reconhecimento automático das línguas de sinais;
2. Construir uma base de sinais da Libras, com protocolo de gravação definido e reprodutível, disponibilizando-a para a comunidade científica;
3. Elaborar uma análise descritiva da base de dados criada para facilitar o entendimento dos dados disponibilizados, além de auxiliar o seu pré-processamento para a

classificação dos sinais;

4. Apresentar uma metodologia de classificação dos sinais da Libras utilizando técnicas de Aprendizado Profundo (*Deep Learning*); e

5. Comparar as abordagens utilizando os vídeos dos sinais e os pontos do corpo, buscando a metodologia mais independente do sinalizador.

1.3 Principais Contribuições da Tese

A metodologia desenvolvida iniciou-se com uma revisão bibliográfica, passando pela criação da base de dados MINDS-Libras até a classificação dos sinais com técnicas de Aprendizado Profundo. Dessa forma, as principais contribuições desta pesquisa foram:

1. Criação e disponibilização pública da base de dados MINDS-Libras (Minds, 2019), juntamente com seu protocolo de gravação e uma análise descritiva dos seus dados;

2. Definição de diretrizes para a construção de bases da língua de sinais;

3. Revisão da literatura apresentando o contexto histórico do reconhecimento da língua de sinais;

4. Desenvolvimento de uma metodologia para o reconhecimento de sinais da Libras independente do sinalizador, utilizando diferentes fontes de dados; e

5. Publicação do artigo *Development and Validation of a Brazilian Sign Language Database for Human Gesture Recognition* (Rezende et al., 2021) que divulga a base de dados MINDS-Libras e apresenta uma metodologia inicial para validação dos dados.

1.4 Estrutura do Documento

Este texto está organizado da seguinte forma:

- O Capítulo 2 - Revisão Bibliográfica apresenta uma breve caracterização da Libras, destacando as principais peculiaridades da língua e o estado da arte na área de reconhecimento da língua de sinais, retratando a sua evolução histórica;

- O Capítulo 3 - MINDS-Libras Dataset descreve as principais diretrizes para a criação de uma base da língua de sinais, juntamente com o protocolo de gravação da MINDS-Libras e uma análise descritiva dos dados disponibilizados;

- O Capítulo 4 - Reconhecimento de Sinais da Libras expõe as abordagens utilizadas para classificar os sinais. Duas diretrizes foram adotadas: uma utilizando os vídeos e a CNN 3D e outra com os pontos do corpo classificando com a TCN;
- O Capítulo 5 - Resultados detalha o efeito de cada abordagem testada para a classificação dos sinais neste trabalho;
- O Capítulo 6 - Conclusões discute a metodologia proposta, apresentando as conclusões, as propostas de continuidade e as publicações resultantes desta pesquisa;
- O Apêndice A - Parâmetros Fonológicos da Língua de Sinais apresenta a estrutura fonológica da língua de sinais e as variações de cada parâmetro;
- O Apêndice B - Descrição dos Sinais Gravados descreve os sinais da Libras selecionados para compor a base de dados criada neste trabalho: MINDS-Libras;
- O Apêndice C - Publicações lista as publicações submetidas a eventos e periódicos da área de Inteligência Computacional e que contribuíram para o desenvolvimento deste estudo. Foram destacados também os eventos *onlines* nos quais esta pesquisa foi divulgada e trabalhos complementares que foram realizados em paralelo a este estudo.

Capítulo 2

Revisão Bibliográfica

Entender a Língua Brasileira de Sinais (Libras) e realizar o seu reconhecimento de forma automática requer um estudo sobre as características da mesma e um domínio das técnicas computacionais que possibilitam realizar tal tarefa. A Libras é uma língua, faz parte de um sistema completo e complexo de comunicação, com todas as peculiaridades presentes em qualquer outro idioma, se diferenciando das línguas orais pela sua natureza visual-gestual, justamente o que a torna objeto deste estudo.

Para o entendimento do cenário em que esta tese se encontra, este capítulo apresenta na Seção 2.1 um breve contexto histórico da Libras, juntamente com os seus principais parâmetros e características que envolvem o aprendizado da língua. Os conceitos foram apresentados com o intuito de esclarecer as principais particularidades dessa forma de comunicação. Para complementar essa seção, o Apêndice A detalha as unidades formacionais de um sinal, também chamadas de parâmetros fonológicos. Em seguida, a Seção 2.2 apresenta trabalhos relacionados ao reconhecimento automático da língua de sinais, com o foco na evolução dessa linha de pesquisa. A análise crítica de diversos estudos possibilitou a escolha das técnicas utilizadas nesta pesquisa e fundamentou, do ponto de vista teórico e empírico, as abordagens implementadas.

2.1 Breve Contexto da Língua Brasileira de Sinais

A partir do Segundo Império é que se começou a falar sobre a língua de sinais no Brasil, com a chegada do educador francês Hernest Hurt, que veio ao país a pedido de Dom Pedro II, pois esse tinha um neto surdo (Instituto Prominas, 2017). Ele trouxe consigo o alfabeto manual[1] francês e a Língua Francesa de Sinais (FSL), dando origem à Libras, que teve como principal influência a FSL.

Um dos principais marcos na história da Libras foi o seu reconhecimento oficial

[1] Alfabeto manual: representação nas línguas de sinais, por meio das mãos, das letras dos alfabetos e os números das línguas orais escritas. Veja a Figura 70 no Apêndice A.

por meio da Lei n^o 10.436/2002 (Brasil, 2002) e sua regulamentação pelo Decreto n^o 5.626/2005 (Brasil, 2005). A legalização da língua foi um importante avanço para a comunidade surda de forma a ampará-la e fornecer suporte para que seus membros pudessem conviver na sociedade conquistando seus espaços. A lei estabelece vários critérios para a inclusão e a acessibilidade dos surdos e define a Libras como:

> "(...) a forma de comunicação e expressão, em que o sistema linguístico de natureza visual-motora, com estrutura gramatical própria, constituem um sistema linguístico de transmissão de ideias e fatos, oriundos de comunidades de pessoas surdas do Brasil."(Brasil, 2002)

De acordo com a lei, a Libras é uma linguagem natural com toda a complexidade dos sistemas linguísticos, servindo de suporte à comunicação, socialização e ao pensamento, com gramática independente da língua oral. Sua estrutura linguística engloba os níveis fonético, fonológico, morfológico, semântico, sintático e pragmático, adaptando os conceitos já utilizados na língua falada. Como o foco deste trabalho é o reconhecimento do sinal, os parâmetros fonológicos[2] serão abordados ao longo desta pesquisa.

A natureza das línguas de sinais faz, muitas vezes, com que as pessoas a confundam com a mímica. Ferreira-Brito (1993) declara que o fato da língua de sinais ser visual-gestual favorece a representação de alguns sinais de forma icônica, entretanto há sinais arbitrários na língua. Mesmo que as línguas de sinais possuam mais essa característica quando comparada com as orais, vale ressaltar que a mímica tenta demonstrar o objeto enquanto o sinal representa o símbolo que foi convencionado para o objeto em questão (Gesser, 2009).

A menor unidade significativa das línguas de sinais é o sinal. Para determinar seu significado é importante identificar a localização das mãos em relação ao corpo (Ponto de Articulação - PA), a orientação das palmas das mãos (Or), a configuração das mãos (CM), o movimento (M) realizado por elas e as expressões não-manuais (ENM). Para exemplificar esses parâmetros, definidos por Stokoe (1960), a Figura 2 ilustra como o sinal "furacão" é executado. Nesse exemplo a configuração da mão assume a forma da letra "D", o ponto de articulação é ao lado do corpo, o movimento que a mão realiza é espiral para cima com a palma da mão orientada para a esquerda e a expressão facial é caracterizada pela testa franzida. Realizando um paralelo com a língua oral, um sinal pode ser comparado com uma palavra e o movimento com o som delas.

Em relação ao processo de aprendizado da Libras, ele se dá como em qualquer outra língua, em níveis básico, intermediário e avançado. Os estímulos oral-auditivo-visual-

[2] O termo fonologia se refere não somente ao estudo dos sons da língua, mas também às unidades mínimas que compõem os sinais. Em 1960, o linguista americano Stokoe (Stokoe, 1960) propôs o termo quirologia, do grego, estudo das mãos, entretanto, a fim de estabelecer um paralelo com as línguas orais, o termo fonologia continua sendo adotado pelos pesquisadores (Quadros e Karnopp, 2004). Cada parâmetro fonológico possui uma gama de categorias e as mesmas foram detalhadas no Apêndice A.

Palavra ⇔ Sinal
Sons ⇔ Movimentos

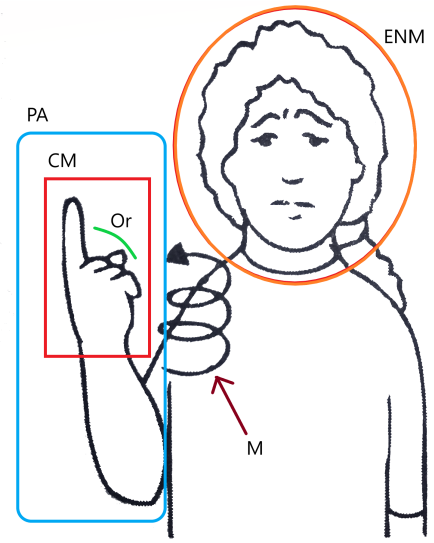


Figura 2 – Parâmetros fonológicos da Libras: Ponto de articulação (PA), Configuração de mão (CM), Movimento (M), Orientação da palma da mão (Or) e Expressões não-manuais (ENM).
Fonte: Rezende et al. (2021).

gestual utilizados em seu aprendizado dependem da idade do aluno, sua motivação, tipo de ensino e se ela será sua língua materna (L1) ou segunda língua (L2). O processo para ser fluente depende dos vários fatores citados e de uma imersão na cultura como um todo. Dessa forma, é importante ressaltar que o tamanho do vocabulário adquirido não é diretamente proporcional à fluência no idioma, além de não ser possível mensurar um número exato de sinais que permita uma pessoa ser fluente na língua. No cenário apresentado pelos dicionários Capovilla et al. (2017a,b,c), há 14500 sinais documentados com seus respectivos verbetes. Os sinais são incorporados à língua à medida em que os surdos se propõem a realizar tal tarefa e difundi-la na comunidade. Isso significa que a criação de novos sinais cabe, exclusivamente, aos surdos.

Por fim, vale frisar que as características descritas nesta seção valem para todas as línguas de sinais. Embora possa existir um histórico de influências entre elas como, por exemplo, entre a Libras e a FSL, há questões culturais que são próprias de cada lugar, o que tornam cada idioma único. De acordo com Simons e Fennig (2018) há 142 línguas de sinais distribuídas nos 108 países destacados na Figura 3. Isso mostra que cada comunidade linguística tem a(s) sua(s) própria(s) língua(s) e não há uma que seja universal.

Figura 3 – Distribuição das línguas de sinais no mundo.
Fonte: Rezende et al. (2021).

2.2 Reconhecimento Automático da Língua de Sinais

A língua de sinais é uma forma de comunicação que utiliza os sentidos visual-gestual para a exposição do pensamento. Nesse caso os gestos tornam-se uma forma de transmissão de ideias, de expressar uma mensagem, e estão presentes na maioria das atividades cotidianas, complementando ou substituindo a fala. Dada a sua importância nas atividades humanas, verifica-se um grande interesse pelas comunidades de Visão Computacional e Aprendizado de Máquina (*Machine Learning*) em analisar esses movimentos corporais a partir de dados visuais (Escalera et al., 2017), desenvolvendo a habilidade computacional da máquina para entendê-los e, quando necessário, tomar decisões baseadas neles. Surgiu, dessa forma, a área de reconhecimento computacional de gestos (*Gesture Recognition*) com o intuito de identificar automaticamente quaisquer gestos humanos.

Dentre os mais variados objetivos que cada sistema de reconhecimento de gestos possui, há uma arquitetura básica que serve de referência para a área, como apresenta a Figura 4. De acordo com Hasan e Misra (2011), essa arquitetura é dividida nas três fases principais:

1. Pré-processamento de imagens: que dá ênfase na remoção de ruído, *background* e outros objetos não relacionados com o estudo. Um passo importante nessa etapa é a segmentação da região de interesse (ROI);

2. Extração de características: responsável em converter a ROI extraída em uma versão

numérica, possibilitando que o computador a processe e a compreenda; e

3. Algoritmo de reconhecimento: usado para estabelecer uma técnica para encontrar o que aquele gesto representa dentre os gestos armazenados em uma base de dados.

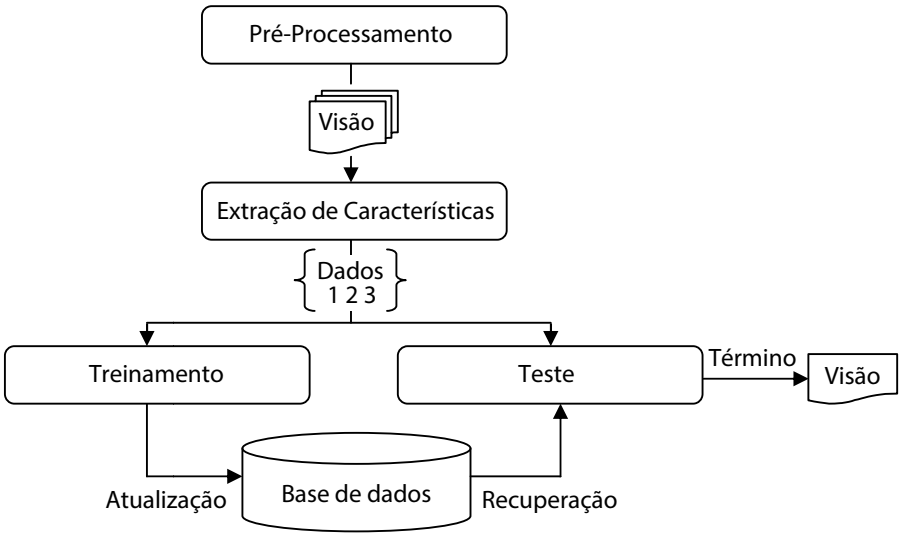


Figura 4 – Etapas básicas de um sistema de reconhecimento de gestos.
Adaptado de: Hasan e Misra (2011).

De acordo com Escalera et al. (2017), há registros de estudos na área de reconhecimento de gestos desde a década de 80, realizando tarefas de reconhecimento de gestos estáticos (Cui e Weng, 2000), detecção de partes do corpo (Chen et al., 2003), reconhecimento de ações e atividades (Li et al., 2010) e reconhecimento da língua de sinais (SLR). Segundo Elakkiya e Selvamani (2017), os gestos das línguas de sinais são considerados como os mais importantes a serem reconhecidos na hierarquia da área. Tal mérito se deve ao fato do reconhecimento automático de sinais ser uma pesquisa em desenvolvimento que inclui áreas como Visão Computacional, Redes Neurais e Aprendizado de Máquina, e por estar diretamente ligada a um sistema de comunicação utilizado por pessoas que possuem algum nível de deficiência auditiva (Elakkiya e Selvamani, 2017).

Os estudos nessa linha de pesquisa iniciaram-se nos anos 80 com Tamura e Kawasaki (1988). Os autores apresentaram a ideia de reconhecimento por meio dos parâmetros manuais da língua, o que é uma característica presente em, aproximadamente, 98% dos trabalhos publicados[3] nessa área. De fato, a análise das informações relativas às mãos é uma tendência, pois elas trazem as principais características para identificar um sinal computacionalmente, devido aos parâmetros manuais (CM, M, Or e PA) serem as unidades formacionais de um sinal, juntamente com as ENM's.

[3] As análises apresentadas nesta seção são resultados de um estudo dos trabalhos publicados até maio de 2021 nos principais periódicos e conferências da área de Aprendizado de Máquina e Visão Computacional.

A posteriori, surgiram novas tendências em relação a aquisição dos dados. Nessa etapa é necessário investigar os sensores e dispositivos utilizados, como os dados serão disponibilizados e em qual formato, além das técnicas que serão aplicadas na etapa de pré-processamento dessas informações. Diante desses fatores, destaca-se o trabalho de Starner et al. (1997, 1998) que desenvolveram um sistema em tempo real para capturar os sinais por meio de uma estrutura acoplada na cabeça do sinalizador. Nesse caso, tem-se uma representação do sinal pela visão de quem o executa, como mostra a Figura 5. Apesar do experimento ser dependente da textura por rastrear a mão pela cor, a visão de primeira pessoa forneceu informações suficientes para o reconhecimento da Língua de Sinais Americana (ASL).

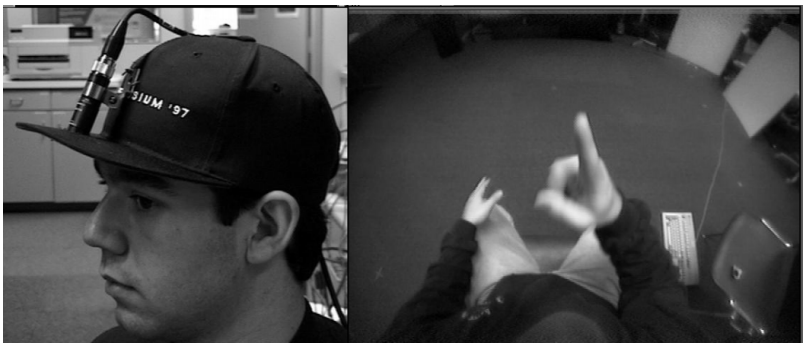


Figura 5 – Sistema de aquisição dos sinais acoplado na cabeça.
Fonte: Starner et al. (1998).

Nessa mesma linha, Rao e Kishore (2018) realizaram a aquisição dos dados por meio da câmera frontal de um celular, como ilustra a Figura 6, apresentando uma abordagem diferente na captura dos sinais por não necessitar de um estúdio de gravação controlado e por utilizar uma tecnologia já difundida e de fácil acesso. Tanto esse trabalho quanto o de Starner et al. (1997, 1998), citados anteriormente, e Thiracitta et al. (2021) exemplificam possibilidades no âmbito das ferramentas para a aquisição dos dados. Não há fatores que limitem os sensores a serem utilizados, a não ser os próprios direcionamentos dados pela pesquisa em questão.

Figura 6 – Sistema de aquisição dos sinais utilizando um celular.
Fonte: Rao e Kishore (2018).

Falar da aquisição dos dados também remete a um problema inerente da área de reconhecimento da língua de sinais que é o acesso a uma base de dados disponível publicamente para treinamento de sistemas. Desde o trabalho de Tamura e Kawasaki (1988), que realizaram o reconhecimento da Língua Japonesa de Sinais (JSL), aproximadamente 55% dos estudos criaram as suas próprias bases de dados e realizam o reconhecimento de uma parcela de sinais da sua língua. Nessa mesma linha, Athira et al. (2019) e Thiracitta et al. (2021) relataram que na Língua Indiana (ISL) e Indonésia (SIBI) de Sinais não há um conjunto de dados padrão disponíveis para classificação. Essa falta de base de dados mostra que independente da língua e mesmo 30 anos após a publicação dos primeiros trabalhos da área, reconhecer sinais automaticamente ainda se torna um desafio e precisa de um esforço da comunidade científica para mudar tal realidade. Esse não é um problema inerente apenas à JSL e à ISL. Isso é um fato presente em praticamente todas as línguas de sinais. Diante dessa realidade, este trabalho apresentará no Capítulo 3 algumas bases de dados disponíveis na literatura, as características desejáveis para esse tipo de conjunto de dados, além de um descritivo de como foi criada a base de dados de sinais da Libras desta pesquisa, chamada de MINDS-Libras. Esse conjunto preenche parte de uma lacuna da área de reconhecimento de sinais[4], mas que ainda precisa de atenção para que não sejam criados conjuntos de dados não representativos e não escalonáveis.

Após os anos 90, houve um crescimento em relação ao número de trabalhos publicados na área, como mostra a Figura 7. Muito dessa evolução se deve ao surgimento de ferramentas de Visão Computacional, como o *OpenCV* (Intel Corporation, 2019) no ano 2000, que permitem a extração de informações das imagens e a interpretação das mesmas, propiciando o desenvolvimento da capacidade de percepção da máquina. O processamento de imagens é uma das principais tarefas dos sistemas de reconhecimento automático e que pode ser feito de forma eficiente utilizando tais ferramentas. Com elas é possível remover o ruído, aplicar técnicas de correlação, detectar regiões de interesse, reamostrar dados, entre outros. Dessa forma, a imagem é preparada para as próximas etapas do sistema de reconhecimento e, consequentemente, para a classificação. Aplicando muitos desses conceitos, uma quantidade significativa de trabalhos realizaram o reconhecimento automático de sinais estáticos[5] como, por exemplo, o reconhecimento do alfabeto manual[6] (Viéville e Crahay, 2004; Assaleh e Al-Rousan, 2005; Ibarguren et al., 2010; Yoon et al., 2012; Elons et al., 2013; Trigueiros et al., 2015; Cambuim et al., 2016; Jadooki et al., 2017; Jimenez et al., 2017; Lee e Lee, 2017; Quesada et al., 2017; Kakoty e Sharma, 2018; Joy et al., 2019; Rho et al., 2020; Sahana et al., 2020; Adithya e Rajesh, 2020b; Tanwar et al., 2019; Shah et al., 2021; Lee et al., 2021).

[4] Neste trabalho, o termo "reconhecimento de sinais" se refere ao reconhecimento automático de sinais da língua de sinais.

[5] Sinais estáticos são sinais sem movimento durante a execução do sinal.

[6] Alfabeto manual é composto por letras e números. Dentre as letras, vale ressaltar que "h", "j", "x" e "z" possuem movimento e, com isso, são classificadas como sinais dinâmicos.

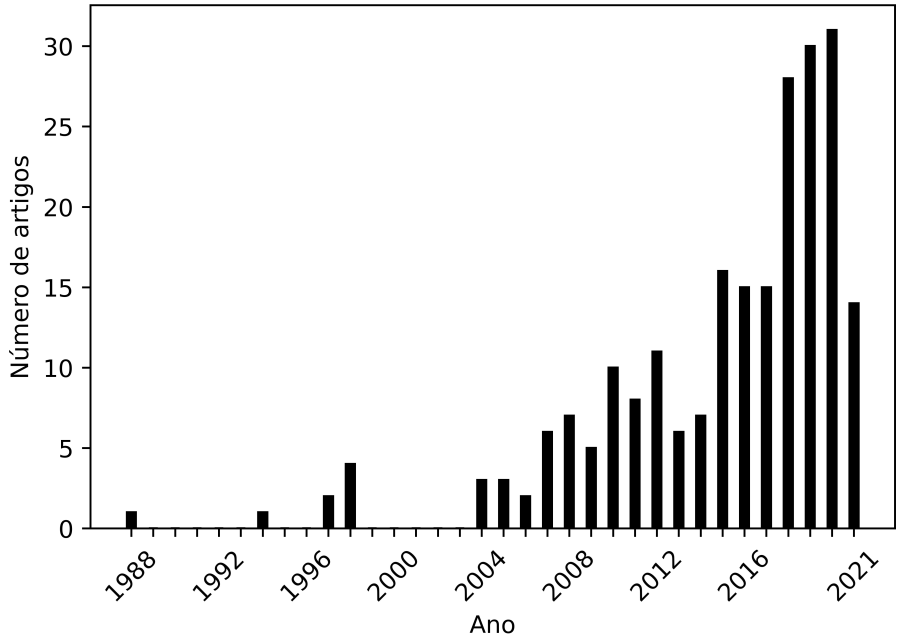


Figura 7 – Histórico de publicações que realizaram o reconhecimento automático de sinais (Dados coletados até maio/2021).

Além do reconhecimento do alfabeto, foi no início dos anos 2000 que começaram a surgir trabalhos que utilizavam as informações faciais no seu sistema de reconhecimento (Holden et al., 2005; Infantino et al., 2007; Nayak et al., 2008; Yang et al., 2008; Vogler e Goldenstein, 2008; Al-Rousan et al., 2009; Aran et al., 2009b; Papapetrou et al., 2009; Yang e Lee, 2010; Kelly et al., 2010; Mohandes et al., 2012; Hadfield e Bowden, 2013; Koller et al., 2015; Shohieb et al., 2015; Fagiani et al., 2015; Roh e Lee, 2015; Elakkiya e Selvamani, 2017; Kumar et al., 2018a,b,c; Kishore et al., 2018; Kumar et al., 2018d,e; Ibrahim et al., 2018; Rao e Kishore, 2018; Elakkiya e Selvamani, 2018; Kumar et al., 2018f; Ravi et al., 2019; Elakkiya e Selvamani, 2019; Koller et al., 2019; Xiao et al., 2020a; Raghuveera et al., 2020; Elakkiya et al., 2021). O rosto e suas expressões fazem parte de um dos parâmetros fonológicos da língua de sinais, chamado de expressões não-manuais (ENM), que também é caracterizado pelo movimento corporal. Apesar das configurações manuais serem mais significativas ao tentar reconhecer um sinal, os dados faciais possuem informações relevantes e discriminativas para tal aplicação. Esses trabalhos muitas vezes utilizaram a face como referência espacial e outros extraíram pontos faciais para compor o vetor de características. Um exemplo da importância da expressão facial no reconhecimento dos sinais é o estudo realizado por Infantino et al. (2007) que utilizaram o movimento labial para segmentar vídeos e, posteriormente, reconhecer os sinais isoladamente. Esse trabalho trata da Língua Italiana de Sinais e uma das suas peculiaridades é o fato do sinalizador movimentar os lábios ao executar o sinal. Com isso, quando não há movimento labial, não há sinal sendo executado e é possível realizar a segmentação da frase, como ilustra a Figura 8. Essa característica não está presente na Libras, tornando a tarefa

de fragmentação de vídeos um desafio na área. Na língua brasileira, os movimentos de finalização de um sinal são ligados ao início do sinal que será executado em seguida.

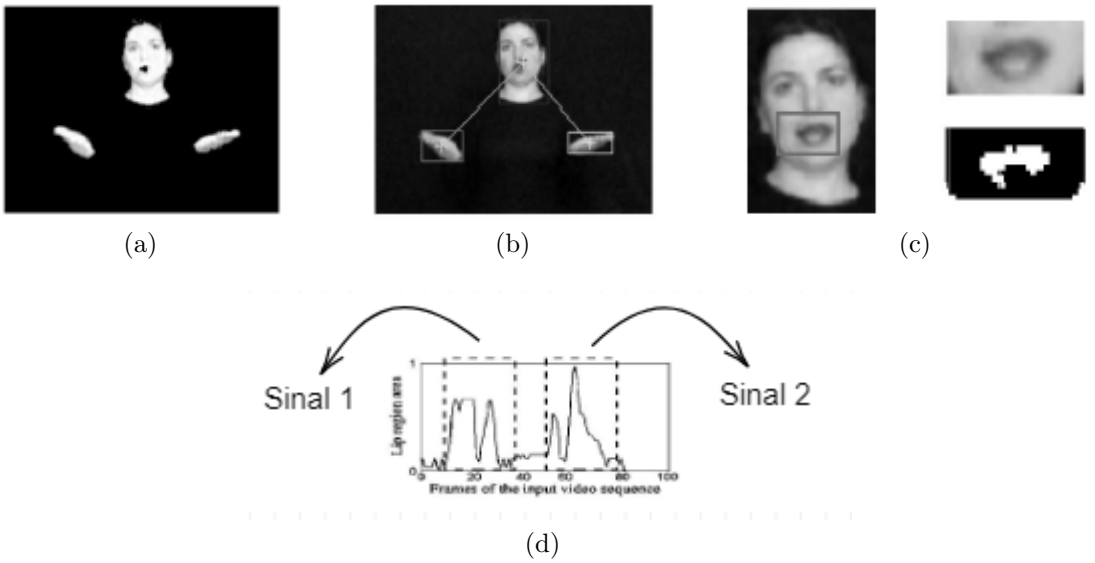


Figura 8 – Segmentação de sinal por meio do movimento labial: (a) detecção de movimento, (b) detecção das mãos e lábios, (c) exemplo de extração de lábio, e (d) segmentação de uma frase composta por dois sinais, analisando a área dos lábios *versus* os quadros na sequência de um vídeo.
Adaptado de: Infantino et al. (2007).

Como visto anteriormente, há estudos que incluíram a informação facial de forma a complementar os parâmetros manuais. Entretanto, há também uma parcela de trabalhos, cerca de 2%, que utilizaram apenas a face como parâmetro para reconhecer sinais. Muitos desses estudos utilizaram as ENM's com o intuito de (i) reconhecer a natureza do que foi dito como, por exemplo, uma afirmação ou negação ou uma pergunta (Nguyen e Ranganath, 2012; Liu et al., 2014), (ii) analisar os aspectos gramaticais e a função sintática da face (Caridakis et al., 2014) e até (iii) investigar as emoções que estão associadas aos sinais (Kumar et al., 2017). Nessa última linha, Guerra et al. (2018) apresentou um estudo que surgiu da abordagem de Rezende et al. (2017). Ambos utilizaram a mesma base de dados, mas Rezende et al. (2017) classificou os sinais pela expressão facial, não levando em consideração que as ENM's acrescentam informações ao reconhecimento quando consideram-se também as mãos. Isso provocou o reconhecimento de parâmetros espúrios e evidenciou que todos os parâmetros fonológicos são importantes para o reconhecimento automático de sinais, mas há características mais significativas que as outras.

Do ponto de vista das abordagens encontradas na literatura para realizar o reconhecimento da língua de sinais, percebe-se que há uma gama de técnicas e ainda não há uma metodologia ou ferramenta específica que resolva tal problema. São muitos os critérios que precisam ser analisados, sejam eles: a gramática de cada língua, os parâmetros fonológicos, a base de dados, como as características serão extraídas, o que será classificado

e como será classificado. Entretanto, em relação à extração de características, os artigos podem ser divididos em duas classes: uma que utiliza técnicas de Visão Computacional, que foram aplicadas em cerca de 76% dos artigos, e outra que utiliza Sensores acoplados ao indivíduo. O crescimento notável no número de publicações no âmbito da Visão Computacional, após os anos 2000, discutido quando a Figura 7 foi apresentada, agora é quantificado também pela Figura 9. Dentre os fatores que proporcionaram essa evolução, destacam-se: (i) o aumento do poder computacional das máquinas, (ii) a tecnologia não invasiva quando comparada com a utilização de sensores e (iii) a disponibilização e popularização das câmeras em diversos equipamentos.

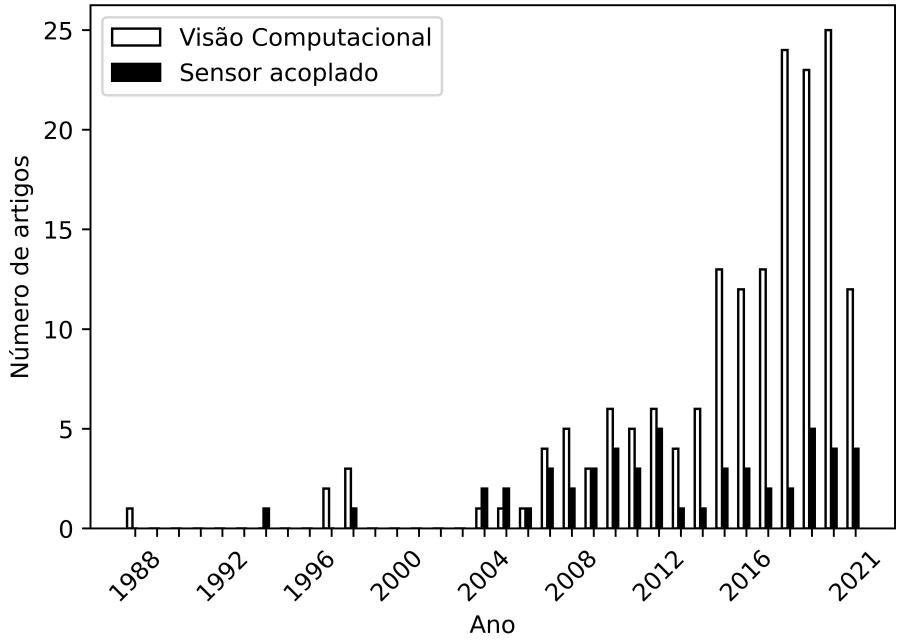


Figura 9 – Número de publicações que utilizaram técnicas de extração de características baseadas em Visão Computacional ou em Sensores acoplados ao indivíduo (Dados coletados até maio/2021).

Em relação à abordagem com Sensores para realizar o reconhecimento automático de sinais, apenas as configurações manuais são consideradas. Encontram-se, na literatura, trabalhos que utilizam luvas e, acoplado a elas, dispositivos como rastreadores de posição (Fang et al., 2004; Gao et al., 2004; Kadous e Sammut, 2005; Fang et al., 2006; Bashir et al., 2006; Zhou et al., 2009; Júnior et al., 2017) e movimento (Oz e Leu, 2011), detectores de aceleração e movimento (Tubaiz et al., 2015), identificadores da forma da mão (Pan et al., 2020a) e trajetória (Kong e Ranganath, 2008, 2014), sensores acelerômetros (Bui e Nguyen, 2007; Ibarguren et al., 2010; Li et al., 2015b; Gałka et al., 2016; Kakoty e Sharma, 2018), sensores de ângulo (Tolba e Abu-Rezq, 1998; Pradhan et al., 2008; Yoon et al., 2012; Mohandes, 2013) e posicionamento dos dedos das mãos (Ahmed et al., 2021). Outros estudos desconsideram as luvas e trabalham apenas com os sensores posicionados ao longo do braço e da mão (Kosmidou e Hadjileontiadis, 2009; Kong e Ranganath, 2010; Kosmidou

e Hadjileontiadis, 2010; Zhang et al., 2011; Kosmidou et al., 2011; Li et al., 2012; Wu et al., 2016; Yang et al., 2016; Lee e Lee, 2017; Zhao et al., 2019; Khomami e Shamekhi, 2021; Gupta e Kumar, 2021).

Mesmo com uma grande variabilidade de dispositivos eletrônicos percebe-se, pela média de publicações, que essa abordagem ainda não se difundiu na área devido a dependência desses aparatos. Alternativamente ao uso de sensores, há uma parte dos trabalhos que utiliza luvas simples como forma de facilitar a detecção das mãos e, em seguida, aplicam técnicas de extração de características baseadas em Visão Computacional (Dorner e Hagen, 1994; Assaleh e Al-Rousan, 2005; Kelly et al., 2010). Nesses casos têm-se sistemas independentes de sensores fixados nas mãos, mas que são dependentes da análise de textura.

Apesar de serem técnicas independentes uma da outra, Hassan et al. (2019) empregaram tanto as informações visuais quanto as sensoriais para compor as características dos sinais, como ilustra a Figura 10. Nesse trabalho foi realizada, também, uma análise de cada uma das informações separadamente e foi constatado que os dados baseados em sensores foram mais precisos do que os dados baseados na visão (Hassan et al., 2019) quando se diz respeito a mensurar informações relativas às mãos. Contudo, a utilização de luvas instrumentalizadas é invasiva, compõe um sistema mais caro e, muitas vezes, torna-se inviável. Portanto, os autores concluíram que é factível investir em técnicas de visão.

Figura 10 – Coleta de dados utilizando técnicas de Visão Computacional e Sensores acoplado às mãos.
Fonte: Hassan et al. (2019)

O progresso da área de reconhecimento automático das línguas de sinais se deve não só à Visão Computacional, mas também ao Aprendizado de Máquina e ao Processamento de Imagens, áreas que se comunicam e são utilizadas para resolver problemas variados da SLR. Como ponto em comum dessas três áreas da Inteligência Computacional, encontra-se o Aprendizado Profundo (*Deep Learning*). Esse termo representa um conjunto de técnicas de Aprendizado de Máquina que surgiram com o intuito de suprir uma lacuna das Redes Neurais Artificiais (RNA) quando se tem uma grande quantidade de dados e vetores de características extensos. Seus modelos computacionais são compostos de múltiplas

camadas de processamento, para aprenderem com os dados em alto nível de abstrações (LeCun et al., 2015; Guo et al., 2016). O Aprendizado Profundo passa por todos os campos da Inteligência Artificial (Skansi, 2018) e se popularizou na resolução de problemas de visão (Krizhevsky et al., 2012a). Diante da grande potencialidade dessa técnica, ela foi a escolhida para reconhecer os sinais da Libras que serão abordados nesta pesquisa.

A Figura 11 quantifica os artigos que aplicaram Aprendizado Profundo para resolver o problema de reconhecimento das línguas de sinais utilizando Visão Computacional (Li et al., 2015a; Inoue et al., 2015; Kim et al., 2017; Kumar et al., 2018d; Huang et al., 2018c; Kumar et al., 2018e; Islam et al., 2018; Li et al., 2018; Koller et al., 2018; Xiao et al., 2018; Joy et al., 2019; Liao et al., 2019; Ravi et al., 2019; Tyukin et al., 2019; Nakjai e Katanyukul, 2019; Lim et al., 2019; Ferreira et al., 2019; Sharma et al., 2020; Lee et al., 2021). Percebe-se que esta é uma tendência nas aplicações devido ao desempenho da técnica, que não é utilizada apenas no reconhecimento de sinais, mas também em reconhecimento facial, detecção de objetos em imagens e extração de características.

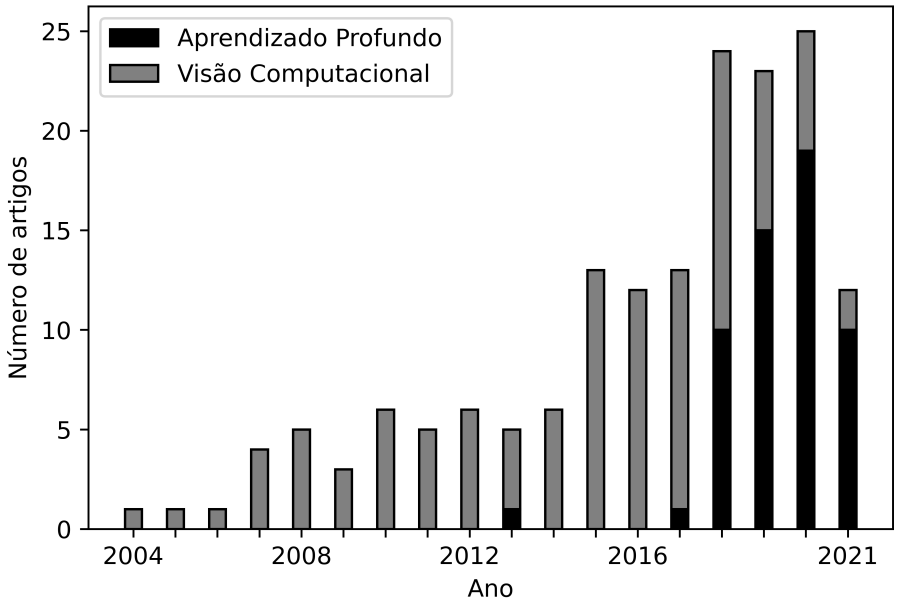


Figura 11 – Número de publicações que aplicaram o Aprendizado Profundo nas pesquisas de Visão Computacional (Dados coletados até maio/2021).

Existe uma variedade de métodos de Aprendizado Profundo na literatura que têm se destacado por fazerem com que as máquinas processem e aprendam informações a partir de dados em diversos formatos, como ilustra a Tabela 1. Isso se torna possível porque essa abordagem expande a quantidade de camadas intermediárias de uma RNA, atribuindo tarefas específicas e progressivas a cada uma delas. Nesse contexto, a Rede Neural Convolucional (CNN) é um método que ganha destaque nesta pesquisa (i) por ter um processo de aprendizado supervisionado, (ii) por possuir uma arquitetura de múltiplas camadas que permite que ela lide com dados de entrada dispostos em vários arranjos (LeCun

et al., 2015), (iii) por ser uma topologia que considera a estrutura espacial dos dados (Guo et al., 2018) e (iv) por ser capaz de obter informações temporais deles (Tran et al., 2014). Conforme apresentado na Figura 11, o Aprendizado Profundo tem se popularizado ao longo dos anos dado o aumento expressivo do poder computacional das máquinas e, nesse contexto, as CNN's seguem a mesma tendência para o reconhecimento da língua de sinais (Huang et al., 2015; Sincan e Keles, 2020; Al-Hammadi et al., 2020a,b; Huang e Ye, 2021a; Qi et al., 2020; Rastgoo et al., 2020; Sharma et al., 2020; Rezende et al., 2021; Thiracitta et al., 2021; Jain et al., 2021; Hisham e Hamouda, 2021).

Tabela 1 – Trabalhos que aplicaram técnicas de Aprendizado Profundo.

Trabalho	Língua de Sinais	Técnica de Solução
Huang et al. (2018a)	Chinesa e Italiana	CNN 3D
Kumar et al. (2018e)	Indiana	CNN
Kumar et al. (2018d)	Indiana	ResNet e CNN
Islam et al. (2018)	Bangladesh	CNN
Li et al. (2018)	Americana	CNN
Xiao et al. (2018)	Chinesa	CNN e DBN
Ferreira et al. (2019)	Americana	CNN
Xiao et al. (2019)	Chinesa	LSTM e HMM
Zhang et al. (2019)	Chinesa	D-shift Net e VGG16
Joy et al. (2019)	Indiana	Nasnet e InceptionV3
Liao et al. (2019)	Chinesa	B3D ResNet
Cui et al. (2019)	Alemã	GoogLeNet, VGG-S e Bi-LSTMs
Cui et al. (2019)	Indiana	LSTM
Koller et al. (2019)	Alemã	CNN, LSTM e HMM
Guo et al. (2019)	Chinesa	LSTM
Zhao et al. (2019)	Americana	ResNet
Ravi et al. (2019)	Indiana	CNN
Tyukin et al. (2019)	Americana	CNN
Al-Hammadi et al. (2020b)	Americana e Arábica	CNN 3D
Aly e Aly (2020)	Arábica	BiLSTM
Papastratis et al. (2020)	Chinesa	CNN 2D e BLSTM
Xiao et al. (2020a)	Chinesa e Alemã	CNN e LSTM
Pan et al. (2020b)	Chinesa	BLSTM
Kumar et al. (2020)	Indiana	CNN
Xiao et al. (2020b)	Chinesa	Bi-LSTM
Qi et al. (2020)	Indiana	CNN
Adithya e Rajesh (2020b)	Americana	CNN
Jiang et al. (2020)	Chinesa	CNN
Thiracitta et al. (2021)	Indonésia	CNN
Suneetha et al. (2021)	Indiana	M2DA-NET
Bencherif et al. (2021)	Arábica	CNN 3D
Xu et al. (2021)	Chinesa	VGG net e LSTM
Huang e Ye (2021b)	Chinesa	LSTM

Em relação às línguas de sinais, há aproximadamente 142 no mundo, distribuídas

em 108 países incluindo o Brasil, como mencionado na Seção 2.1. Entretanto, a Figura 12 mostra que o número de publicações na área de reconhecimento da Libras está aquém quando comparado com as línguas de sinais Americana, Indiana, Árabe e Chinesa. Apesar da Libras ser a segunda língua reconhecida no Brasil, são poucos os esforços para que se chegue a uma área consolidada quando se diz respeito às pequisas para o seu reconhecimento automático. A falta de base de dados é um dos principais fatores que impedem a evolução da área e que faz com que trabalhos como Júnior et al. (2017) validem as suas metodologias em outras línguas de sinais. Para ilustrar a pesquisa realizada com a Libras, os trabalhos de Almeida et al. (2014), Cambuim et al. (2016), Cardenas e Chavez (2020) e Cerna et al. (2021) serão destacados.

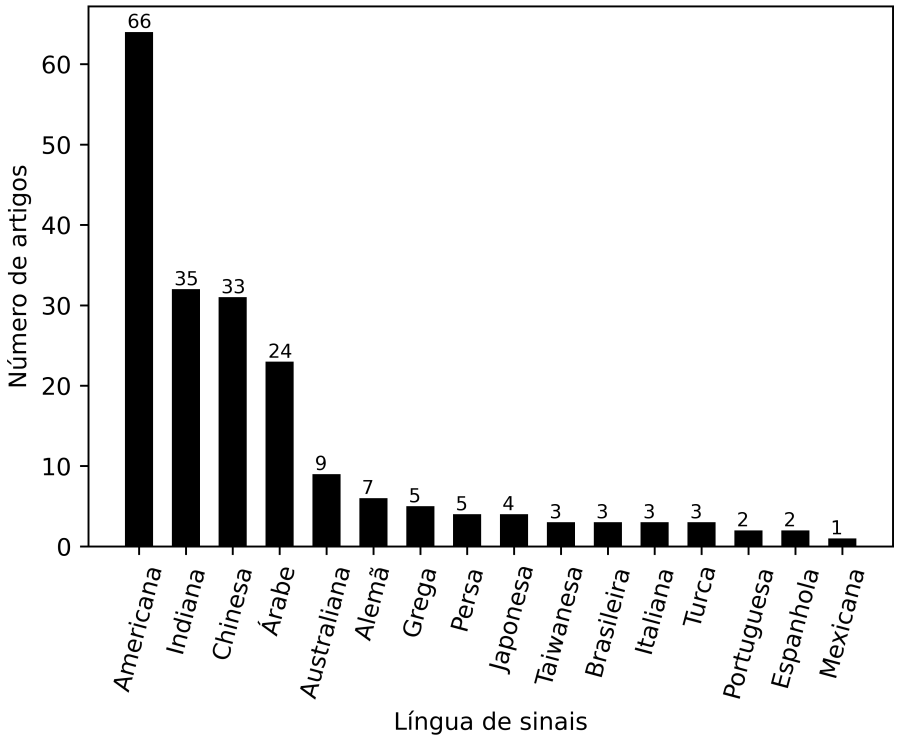


Figura 12 – Número de publicações relacionadas à trabalhos de reconhecimento de sinais em cada língua de sinais (Dados coletados até maio/2021).

Almeida et al. (2014) realizou o reconhecimento de 34 sinais dinâmicos da Libras por meio de seus parâmetros fonológicos manuais. Ao tratar dados em vídeos, esse trabalho utilizou uma técnica de sumarização para obter os quadros mais significativos de cada amostra, tal como em Castro et al. (2019); Pan et al. (2020b); Al-Hammadi et al. (2020a); Rezende et al. (2021). Em relação à extração de características, a abordagem adotada analisou cada parâmetro fonológico individualmente, classificando-os com a Máquina de Vetores de Suporte (SVM) e realizando uma comparação com a saída desejada ao final desse processo. Vale ressaltar que a SVM é uma técnica estado da arte em vários problemas de Reconhecimento de Padrões e se destaca, também, na SLR (Boulares e Jemni, 2012; Sun et al., 2013; Zhao e Martinez, 2015; Kakoty e Sharma, 2018; Raghuveera et al., 2020;

Shah et al., 2021; Khomami e Shamekhi, 2021; Gupta e Kumar, 2021; Jain et al., 2021).

Cambuim et al. (2016) realizaram a classificação de imagens estáticas. Em um primeiro momento eles realizaram o reconhecimento de 18 letras e, posteriormente, de 61 configurações manuais. Nesse caso foi realizada a tradução de símbolos estáticos em símbolos de texto, através de visão computacional, sem o uso de sensores ou luvas de mão. Para isso, o algoritmo *Extreme Learning Machine* (ELM) foi utilizado, sendo uma implementação aplicada também em Imran e Raman (2020) e Katılmış e Karakuzu (2021). No âmbito do reconhecimento do alfabeto (letras e números) e das configurações manuais, a área está consolidada. Percebe-se que os estudos estão evoluindo para a classificação dessas informações num cenário de *backgrounds* complexos, como em Joshi et al. (2020).

Cardenas e Chavez (2020) e Cerna et al. (2021) apresentaram uma metodologia para reconhecimento da Libras independente do sinalizador. Essa abordagem, presente também em Sincan e Keles (2020); Wadhawan e Kumar (2020) e Pan et al. (2020b), destaca a importância de considerar que cada pessoa tem a sua forma própria de executar o sinal, como acontece na forma de falar de cada indivíduo. Isso significa que a maneira como a informação é transmitida na língua de sinais pode apresentar variações para cada pessoa, alterando a duração do sinal, intensidade do movimento e as expressões faciais. Uma contribuição relevante de Cardenas e Chavez (2020) e Cerna et al. (2021) é a disponibilização de uma base de dados de sinais da Libras, chamada de LIBRAS-UFOP, que contém vídeos em RGB-D e informações do esqueleto. Destaca-se nos resultados a informação de que a característica temporal do esqueleto é essencial para melhorar a precisão do modelo (Cardenas e Chavez, 2020), mostrando que a trajetória manual é um dos parâmetros mais significativos para o reconhecimento dos sinais.

Em suma, esta seção apresentou o contexto histórico da área de reconhecimento automático da língua de sinais, expondo a evolução da área ao longo dos anos. O objetivo foi fundamentar esta pesquisa do ponto de vista teórico e empírico, ressaltando que trabalhar com SLR é lidar com as mais diversas propostas metodológicas e técnicas que a Inteligência Computacional engloba, ao mesmo tempo que é necessário entender a língua para tratar as sutilezas gramaticais da mesma. Dessa forma, o presente trabalho desenvolveu uma metodologia capaz de reconhecer automaticamente sinais da Libras por meio de ferramentas de Aprendizado Profundo, além de criar uma base de dados robusta que permitiu a validação da metodologia proposta.

2.3 Principais Contribuições do Capítulo

Este capítulo apresentou as principais características da Língua Brasileira de Sinais e uma revisão bibliográfica dos trabalhos relevantes da área. Compreender as características da Libras permitiu uma análise dos trabalhos relacionados de forma mais

criteriosa, pois a língua possui sutilezas não comuns na língua falada e não presentes na área de reconhecimento de gestos.

Da revisão realizada até maio de 2021 destacam-se as seguintes informações nos principais periódicos e conferências da área de Aprendizado de Máquina e Visão Computacional:

- Aproximadamente 98% dos trabalhos lidam com informações relativas às mãos;
- Cerca de 55% estudos criaram suas próprias bases de dados;
- Em torno de 76% dos artigos utilizam de técnicas de Visão Computacional;
- As publicações utilizando Aprendizado Profundo em 2020 superaram as dos anos anteriores;
- Por volta de 1% dos artigos realizaram o reconhecimento de sinais da Libras; e
- Os parâmetros manuais são os mais significativos para a SLR.

Diante dessas características, esta pesquisa apresenta a criação de uma base de dados no Capítulo 3 com sinais da Libras e um protocolo de gravação bem definido, permitindo a reprodução e a contribuição da comunidade científica. Como esta base disponibiliza, dentre vários dados, vídeos dos sinais e informações como a trajetória manual, o Capítulo 4 descreve duas abordagens para o reconhecimento dos sinais com base em técnicas de Aprendizado Profundo: a Rede Neural Convolucional 3D (CNN3D) e a Rede Convolucional Temporal (TCN).

Capítulo 3

MINDS-Libras Dataset

A proposta deste trabalho é desenvolver uma metodologia que realize o reconhecimento automático de sinais da Libras, aplicando técnicas de Aprendizado de Máquina, Reconhecimento de Padrões e Visão Computacional. Resolver problemas nessas áreas é lidar com as etapas de (i) aquisição dos dados, (ii) segmentação da região de interesse, (iii) extração e seleção de características e (iv) classificação (Pedrini e Schwartz, 2008).

A aquisição de dados é a etapa que reúne os elementos que serão processados pelo sistema. A consolidação desse estágio resulta no que se chama de Base de Dados (*dataset*), isto é, um conjunto de informações estruturadas que relacionam entre si, podendo ser de natureza numérica, imagens ou variações desses. Na literatura existem servidores que hospedam esses dados, como *Kaggle*[1] e *UCI Machine Learning Repository*[2], disponibilizando informações para o uso de toda a comunidade científica. Entretanto, ainda há uma carência de dados que sejam públicos em alguns campos de pesquisa, como é o caso da área de reconhecimento automático da Língua Brasileira de Sinais.

A criação de uma base de sinais em Libras foi necessária neste trabalho para preencher essa lacuna. O objetivo principal foi disponibilizar dados com protocolo de aquisição reprodutível e escalonável, de forma que pesquisadores usufruam desse *dataset* para treinamento e teste de seus modelos, além de poderem contribuir para as pesquisas na área e propiciar a expansão dessa base, chamada de MINDS-Libras.

Sendo assim, este capítulo apresenta na Seção 3.1 algumas bases disponíveis na literatura e um estudo das principais propriedades que um conjunto de dados da língua de sinais deva ter para ser representativo. Posteriormente, a Seção 3.2 expõe as características da MINDS-Libras, abordando: (i) os sinais escolhidos, (ii) os sinalizadores que realizaram a execução dos sinais, (iii) os sensores e software utilizados para a gravação da base, (iv) o cenário onde os sinais foram gravados e (v) os dados disponibilizados para uso da comunidade científica. Por fim, a Seção 3.3 apresenta uma análise descritiva desses dados.

[1] https://www.kaggle.com/datasets
[2] http://archive.ics.uci.edu/ml/index.php

3.1 Bases de Dados disponíveis na Literatura

Wang et al. (2019), Athira et al. (2019), Wadhawan e Kumar (2020) e Azar e Seyedarabi (2020) relatam que na área de reconhecimento automático da língua de sinais há a necessidade de um conjunto de dados consolidado. Essa declaração se baseia no fato de que cada idioma possui suas especificidades e de que os sinais não são exclusivamente mímicas. O ideal, nesse tipo de aplicação, é que as amostras sejam representativas e em proporção significativa. As alternativas que alguns trabalhos encontram para realizar experimentos com técnicas de aprendizado de máquina foram a utilização de bases de dados de gestos, devido a sua grande similaridade com os movimentos executados nas línguas de sinais (Li et al., 2010; Xia et al., 2011; Escalera et al., 2013; Liu e Shao, 2013; Bloom et al., 2016; Ben Tamou et al., 2017) ou utilizar o conjunto de dados de uma língua de sinais que possa estar melhor estruturada, como em Nguyen e Ranganath (2012); Júnior et al. (2017).

Quando se trata da utilização de bases da área, até o momento, cerca de 55% dos trabalhos presentes na literatura criam seus próprios conjuntos de dados. Essa realidade tem como consequência bases de dados que possuem poucas amostras devido às dificuldades inerentes ao processo da sua criação. Isso acaba enfraquecendo a metodologia proposta nos estudos, pois os modelos não adquirem a capacidade de generalizar, tornando-se especializados aos poucos dados que se tem. Além da quantidade de software e equipamentos disponíveis para gravação, outro problema encontrado é que muitas das bases de sinais não possuem documentação que explica como os dados foram coletados e manipulados, dificultando a reprodução e replicação do protocolo utilizado.

Entre os trabalhos que criam as suas próprias bases de dados, algumas delas estão disponíveis ao público, conforme listado na Tabela 2. As principais diferenças desses *datasets* estão relacionadas com o tipo de dado disponibilizado (escala de cinza, RGB, RGB-D, pontos da face e esqueleto), processo de gravação (sequencial ou isolado), região de interesse capturada (corpo, parte superior do corpo ou mãos), cenário de gravação (neutro, eliminado ou alterado) e o número de sinalizadores. Essa variabilidade deixa claro que não há um padrão para a criação de uma base de dados da língua de sinais e, em uma análise mais profunda, verifica-se que cada uma delas foi concebida para atender a uma tarefa específica.

A Língua Americana de Sinais (ASL) é a mais utilizada quando se fala de pesquisas no âmbito de sistemas de reconhecimento automático, como mostrou a Figura 12 apresentada no Capítulo 2, por ser a língua que investigam há mais tempo. Entretanto, essa condição não torna a ASL superior às demais línguas. Já em relação à língua brasileira, percebe-se que é necessário mais atenção no desenvolvimento de sistemas que realizam o seu reconhecimento, principalmente na etapa de aquisição de dados, que é a etapa inicial desse processo, e na divulgação/publicação dessas bases.

Tabela 2 – Bases de Línguas de Sinais publicadas.

Base de Dados	Ano	Língua	#Amostras
Auslan (Kadous, 2002)	2002	Australiana	2565
RWTH BONTON-50 (Zahedi et al., 2005)	2005	Americana	483
RWTH BONTON-104 (Dreuw et al., 2007)	2007	Americana	201
SIGNUM (Agris, 2008)	2008	Alemã	33210
ASLLVD 2008 (Athitsos et al., 2008)	2008	Americana	9800
eNTERFACE'06 (Aran et al., 2009a)	2009	Americana	760
MSR Action 3D (Li, 2017)	2012	Americana	336
A3LIS (Fagiani et al., 2012)	2012	Italiana	147
RWTH PHOENIX Weather (Forster et al., 2012)	2012	Alemã	45760
RGB-D ASL (Conly et al., 2013)	2013	Americana	1113
PSL ToF 84(Oszust e Wysocki, 2016b)	2013	Persa	1680
PSL Kinect 30 (Oszust e Wysocki, 2016a)	2013	Persa	300
Grammatical Facial Expressions Data Set (Freitas et al., 2014b)	2014	Brasileira	225
Libras-34 Dataset (Kinect v1) (Almeida, 2014a)	2014	Brasileira	170
LSA 64(Ronchetti et al., 2016)	2016	Argentina	3200
Libras-10 Dataset (Almeida et al., 2016)	2016	Brasileira	100
DEVISIGN-D (Wang et al., 2016)	2016	Chinesa	6000
SLR Dataset (MCC Lab, 2020)	2016	Chinesa	25000
ASL LEX 2017 (Caselli et al., 2017)	2017	Americana	993
ISLTD2018 (Machine Vision Lab, 2018)	2018	Indiana	1039
ASL Dataset (Avola et al., 2018)	2018	Americana	1200
CSL Video Dataset (Huang et al., 2018b)	2018	China	25000
ArASL2018 (Latif et al., 2019)	2018	Arábica	54096
MINDS-Libras (Minds, 2019)	**2019**	**Brasileira**	**1200**
Italian Sign Language Alphabet (Pacifici et al., 2020)	2020	Italiana	780
DF-WiSLR (Ahmed et al., 2020)	2020	Indiana	2180
ISL words (Adithya e Rajesh, 2020a)	2020	Indiana	824
RKS-PERSIANSIGN (Rastgoo et al., 2020)	2020	Persa	10000
LIBRAS-Ufop (Cardenas e Chavez, 2020)	2020	Brasileira	2800
AUTSL (Sincan e Keles, 2020)	2020	Turca	38336
WLASL (Li et al., 2020)	2020	Americana	2000

Destacando as bases referentes à Língua Brasileira de Sinais, a Tabela 2 apresenta 4 conjuntos, além da MINDS-Libras criada neste trabalho: *Grammatical Facial Expressions Data Set* (Freitas et al., 2014b), Libras-34 (Almeida, 2014a), Libras-10 (Almeida et al., 2016) e LIBRAS-Ufop (Cardenas e Chavez, 2020). Como esse campo de pesquisa está em desenvolvimento e há muitos subproblemas para serem resolvidos, a descrição de cada uma delas explicita suas características e o objetivo para o qual cada uma foi criada:

- ***Grammatical Facial Expressions Data Set***: Esse conjunto de dados é composto por 18 vídeos gravados com o sensor *Microsoft Kinect v1*. Em cada vídeo, um usuário executa, em frente ao sensor, cinco frases em Libras que requerem o uso de uma expressão facial gramatical, onde cada frase é repetida cinco vezes. São disponibilizados: (a) uma imagem de cada quadro e (b) um arquivo de texto contendo 100 coordenadas (x, y, z) do rosto, como ilustra a Figura 13. As imagens possibilitaram uma rotulagem manual de cada arquivo por um especialista. O foco dessa base é a expressão facial durante a execução de uma frase.

(a) Imagem da face (b) Pontos extraídos

Figura 13 – Exemplo de um quadro e os 100 pontos faciais extraídos de um quadro da base *Grammatical Facial Expressions Data Set*.
Fonte: Freitas (2011)

- **Libras-34 *Dataset* (Kinect v1)**: Essa base foi criada por Almeida (2014b) para realizar o reconhecimento de sinais da Libras com base nos parâmetros fonológicos da língua. Os sinais foram gravados com o *Microsoft Kinect* v1 e o *software* nuiCaptureAnalyze, de forma que foram obtidas gravações simultâneas de vídeos RGB, profundidade e esqueleto, como apresenta a Figura 14. Os 34 sinais que compõem a base foram executados por uma pessoa e repetido cinco vezes. O sinalizador manteve uma distância de aproximadamente 2 metros do sensor, de tal forma que os vídeos gravados enquadram a parte superior do corpo. Com o intuito de destacar as mãos e o rosto do sinalizador, tanto sua vestimenta como o *background* do estúdio eram pretos.

(a) Imagem RGB (b) Profundidade (c) Esqueleto

Figura 14 – Quadros de uma sequência de vídeo da base de dados Libras-34.
Fonte: Almeida (2014b).

- **Libras-10 *Dataset* (Kinect v1)**: Essa base foi criada por Rezende (2016) para análise da expressão facial em reconhecimento de sinais da Libras. Para a criação desse conjunto de dados foi utilizado o sensor RGB-D *Microsoft Kinect* v1 em conjunto com o *software* nuiCaptureAnalyze, capturou simultaneamente imagens RGB, profundidade, pontos referentes ao esqueleto do sinalizador e a 121 pontos da face. A base possui dados de 10 sinais da Libras, gravados 10 vezes cada um por

um único sinalizador, totalizando 100 amostras. O sinalizador se manteve a uma distância aproximada de $1,2$ metros do sensor RGB-D, de forma que somente o seu tronco fosse enquadrado nas imagens geradas. Houve controle de iluminação com foco para a face do sinalizador e a utilização do tecido *Chroma Key*[3] como *background*. Os dados capturados estão dispostos como mostra a Figura 15, em sequência de imagens com lógica temporal.

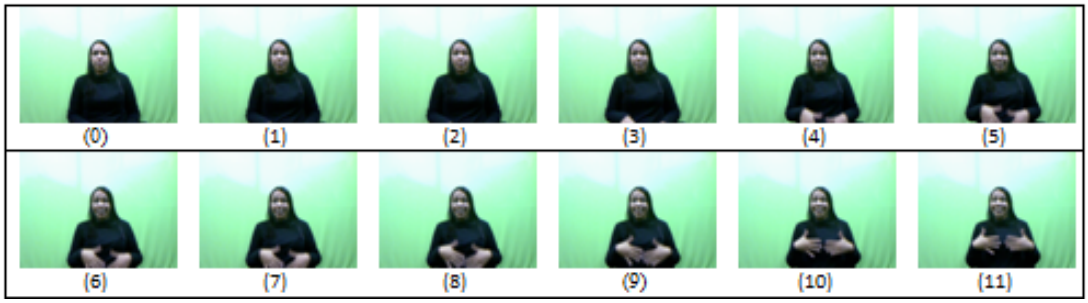
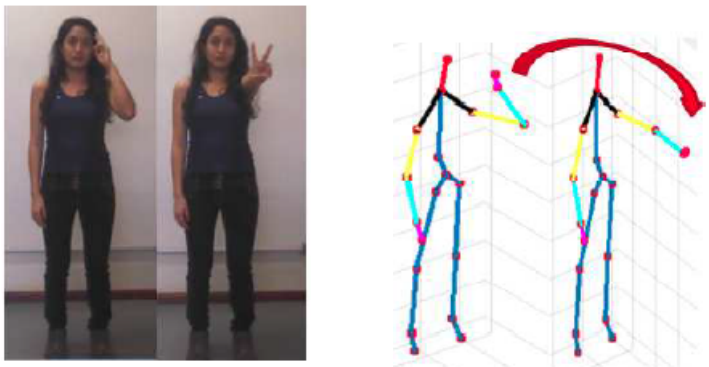


Figura 15 – 12 quadros RGB de uma amostra da base de dados Libras-10.
Adaptado de: Rezende (2016).

- **LIBRAS-Ufop**: O conjunto de dados contém 56 sinais em Libras, realizados por cinco sinalizadores, onde cada um repetiu o sinal 10 vezes, em média. A base contém 2800 sequências de dados, incluindo três modalidades de dados: vídeos RGB, profundidade e posições o esqueleto. Esse conjunto de dados apresenta as seguintes propriedades: (a) sinais com trajetórias semelhantes, mas diferentes configurações de mão e corpo; (b) sinais com configurações de mão e corpo semelhantes, mas trajetórias diferentes; (c) sinais usando uma mão; (d) sinais usando as duas mãos. A Figura 16 exibe um exemplo dos dados disponibilizados nesse *dataset*.



(a) Imagem RGB (b) Esqueleto

Figura 16 – Exemplo de sinal da base de dados LIBRAS-Ufop.
Adaptado de: Cardenas e Chavez (2020).

[3] Chroma Key: técnica corriqueiramente utilizada para posicionar uma imagem sobre uma outra através do anulamento de uma cor sólida, como o verde claro.

Independente da língua que a base de dados representa, diferentes abordagens são utilizadas para realizar o reconhecimento automático desses dados. Isso quer dizer que não há técnica ou ferramenta específica para resolver esse tipo de problema. Assim, muitos critérios precisam ser analisados, como a gramática de cada idioma, parâmetros fonológicos, estrutura da base de dados, extração de características e método de classificação. O que aproxima as línguas de sinais é a modalidade visual-espacial. Ainda assim, as diferenças entre elas indicam que é preciso a análise específica para cada uma.

Investigando os diversos problemas a serem resolvidos na área, viu-se a necessidade de sumarizar as características desejáveis para que um conjunto de dados seja representativo. A Tabela 3 apresenta um estudo que levou em consideração os requisitos necessários para cobrir a maioria das tarefas de aprendizado de máquina que realizam o reconhecimento automático da língua de sinais. Com base na Tabela 2 e nos trabalhos disponibilizados na literatura, os atributos significativos foram: parâmetros fonológicos (CM, PA, M, Or e ENM), natureza do sinal (estático ou dinâmico), elementos (alfabeto, sinal isolado, sentença) que irão compor a base, seu tamanho (número de amostras), número de sinalizadores, cenário (características físicas) e protocolo de gravação.

3.2 Protocolo de Gravação

O protocolo de gravação descrito nesta seção foi construído para garantir a replicabilidade e escalonabilidade da base de dados MINDS-Libras. Essas características fortalecem as pesquisas na área de reconhecimento automático da Língua Brasileira de Sinais, permitindo a validação de sistemas de classificação computacional de forma robusta. Além disso, seguindo as etapas apresentadas, pesquisadores podem contribuir para a expansão da base de dados.

A definição do protocolo de gravação inicialmente apresentado por Almeida (2017) teve como principais referências os estudos realizados por Ruffieux et al. (2014); Almeida (2014b); Rezende (2016), abordando: (i) os sinais escolhidos, (ii) a característica dos sinalizadores, (iii) os sensores e softwares utilizados para aquisição dos vídeos/imagens, (iv) o cenário das gravações e (v) a estrutura dos dados disponibilizados.

Inicialmente, 20 sinais em Libras foram selecionados por uma profissional com proficiência na língua, tendo como critério de escolha a variabilidade dos parâmetros fonológicos, isto é, que houvessem semelhanças e diferenças entre sinais quando se diz respeito à configuração de mão (CM), ao ponto de articulação (PA), ao movimento das mãos (M), à orientação da palma da mão (Or) e às expressões não-manuais (ENM). Cada um desse sinais foi gravado 5 vezes por 12 sinalizadores, totalizando uma base de dados com 1200[4] amostras. O número de gravações de cada sinal foi um valor empírico tendo

[4] A MINDS-Libras contém 20 sinais × 5 repetições × 12 sinalizadores = 1200 amostras.

Tabela 3 – Características desejáveis para uma base de dados da língua de sinais.

Características	Descrição	Referência
Parâmetros fonológicos	O sinal é composto por parâmetros manuais e não-manuais, então é desejável que a base de dados contenha ambas informações. Independente do objetivo do sistema, o *dataset* deve ser completo o suficiente para cobrir o maior número de aplicações possível.	Minds (2019), Dreuw et al. (2007), Oszust e Wysocki (2016a), Cardenas e Chavez (2020), Li (2017), Caselli et al. (2017)
Natureza do sinal	Nas línguas de sinais existem sinais estáticos e dinâmicos. Criar uma base de dados com esses tipos de dados demonstra que a base é representativa em um cenário em que a maioria dos estudos abordam apenas uma das classes.	Minds (2019); Almeida (2014a); Oszust e Wysocki (2016b)
Elementos	Uma base de dados pode contemplar elementos do alfabeto (sinais estáticos), sinais isolados (estáticos ou dinâmicos) ou sentenças, sendo o último considerado o mais completo possível por representar a variação existente no vocabulário da língua e apresentar um contexto entre os sinais. Na literatura os trabalhos lidam apenas com um ou, no máximo, dois desses elementos.	Ahmed et al. (2020)
Tamanho	O tamanho da base de dados é uma variável subjetiva porque não é trivial gravar sinais em grande escala. A busca é por representatividade e proporcionalidade em um universo linguístico. É desejável que os pesquisadores possam contribuir para que a gravação seja feita constantemente e ininterruptamente.	Athitsos et al. (2008); Forster et al. (2012); Ronchetti et al. (2016); Minds (2019)
Número de sinalizadores	A base deve ter mais de um sinalizador executando o sinal para que o sistema de classificação não se especialize nas características físicas do mesmo. Além disso, pequenas variações na execução de sinais podem ocorrer quando ele é executado por diferentes pessoas e o sistema será tão real quanto melhor detectar essas alterações, identificando o sinal e não a pessoa.	Ronchetti et al. (2016); Forster et al. (2012); Minds (2019); Conly et al. (2013); Li (2017)
Cenário de gravação	O controle de iluminação é interessante para que as gravações sejam feitas com qualidade. No entanto, se o sistema for usado em ambientes públicos, a base deve ser registrada com alguma tecnologia que permita a inserção de fundos variáveis, como o *chroma key*. Assim, o classificador poderá aprender cenários diferentes e ser capaz de distinguir o sinal executado.	Almeida et al. (2016); Minds (2019); Almeida (2014a)
Protocolo de gravação	Seguir um protocolo de gravação é a garantia de um padrão nas amostras, permitindo que outros pesquisadores contribuam para a evolução da pesquisa, facilitando a amostragem de mais dados.	Almeida et al. (2016); Minds (2019); Almeida (2014a)

como base a experiência descrita nos trabalhos de Almeida (2014b) e Rezende (2016), além de considerar o espaço requerido para armazenamento desses dados e o tempo de gravação. Em relação aos sinalizadores, havia homens e mulheres, surdos e ouvintes, sem padronização de vestimenta e com características físicas distintas.

Durante a gravação dos sinais, os sinalizadores permaneceram olhando para a câmera em posição de descanso antes e após a execução do sinal, para marcar o início e o fim de cada gravação, como ilustra a Figura 17. Em todas as gravações a posição do sinalizador é fixa, ao centro do vídeo, em pé e ele começa e termina o sinal com as mãos ao lado das pernas. No estúdio de gravação os sensores ficaram em posições fixas, recebendo a mesma quantidade de iluminação e gravando todo o movimento corporal da cintura para cima: expressão facial e o movimento manual.

Figura 17 – Posição de descanso definida para a gravação de cada amostra.

Para a captura dos sinais foi utilizado uma câmera profissional Canon EOS Rebel t5i e o sensor RGB-D *Kinect* v2 para *Xbox One*. Esses dispositivos possibilitaram a disponibilização de dados nas seguintes formas: (i) vídeos dos sinais em RGB, (ii) vídeos com informação de profundidade, (iii) informações de 25 pontos/juntas do corpo[5] e de (iv) 1347 pontos da face[6]. A MINDS-Libras está disponível publicamente em Minds (2019).

3.2.1 Sinais Escolhidos

Nas obras de Capovilla et al. (2017a,b,c), há cerca de 14 mil sinais em Libras que constituem a língua. Entretanto, o fato dela ser visual envolve sutilezas normalmente perceptíveis pelo ser humano dentro de um contexto e que para os sistemas computacionais não é trivial. Diante desta complexidade inerente a uma língua, o trabalho de escolha dos sinais para a construção da base cabe, naturalmente, a um especialista na língua. Como a ideia da base é que ela permite agregar quaisquer sinais, acredita-se que a quantidade de sinais gravados neste momento não seja crítica, pois pode ser ampliada, mas fez-se necessário estabelecer um número inicial para que a metodologia de classificação proposta neste trabalho pudesse ser validada.

[5] As 25 juntas do corpo estão ilustradas na Figura 24, Seção 3.3.2.
[6] Os 1347 pontos da face estão ilustrados na Figura 41, Seção 3.3.3

Com isso, uma profissional com proficiência na Libras selecionou 20 sinais de forma a trazer diversidade para a base em relação aos parâmetros fonológicos da língua. Isso significa que a especialista teve a preocupação em selecionar sinais que proporcionassem a heterogeneidade da base e que também houvessem semelhanças em relação às variáveis CM, PA, M, Or e ENM. Dessa forma, o sinais que compõem a base, representados na Figura 18, são: (a) acontecer, (b) aluno, (c) amarelo, (d) América, (e) aproveitar, (f) bala, (g) banco, (h) banheiro, (i) barulho, (j) cinco, (k) conhecer, (l) espelho, (m) esquina, (n) filho, (o) maçã, (p) medo, (q) ruim, (r) sapo, (s) vacina e (t) vontade. Cada sinal selecionado foi gravado 60[7] vezes, sendo a maior parte deles de natureza dinâmica[8], exceto os sinais "América" (Figure 18d) e "cinco" (Figure 18j). Além do movimento, a mudança na expressão facial também não está presente em todos os sinais[9], pois ela está fortemente relacionada à emoção/sentimento que compõe o seu significado.

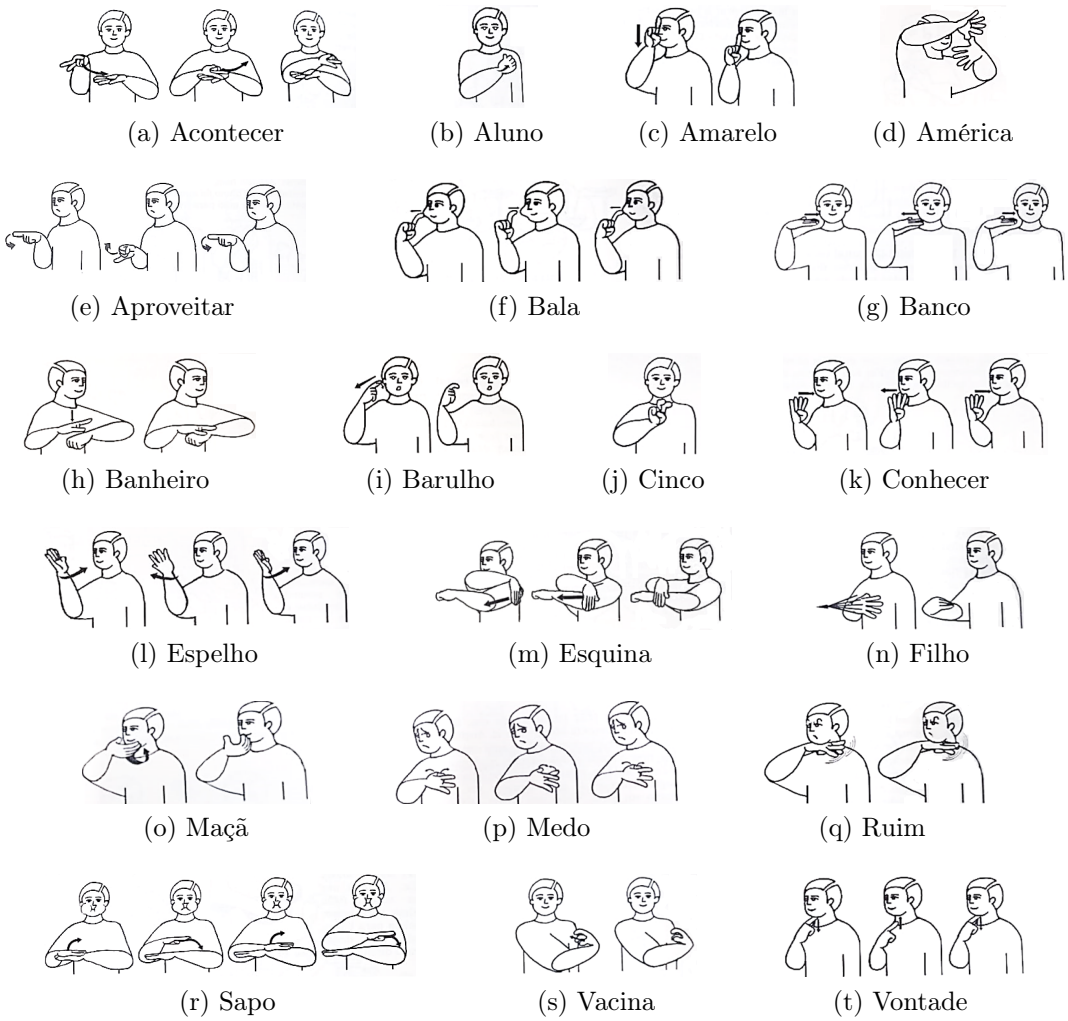


Figura 18 – Sinais que compõem o banco de dados MINDS-Libras.
Fonte: Capovilla et al. (2017a,b,c).

[7] Número de vezes que cada sinal foi gravado: 12 sinalizadores × 5 repetições = 60 gravações.
[8] Dinâmica: termo relacionado com movimento que compõe o sinal.
[9] Veja a Tabela 17 no Apêndice B.

Para exemplificar o critério de escolha dos sinais para a base de dados, Castro (2020) quantificou a variabilidade das amostras agrupando-as de acordo com suas similaridades (algoritmo t–SNE - *T–distributed Stochastic Neighbor Embedding*), como mostra a Figura 19. Utilizando 10 quadros por vídeo, Castro (2020) concluiu que os sinais mais semelhantes possuíam ponto de articulação próximos e os demais parâmetros fonológicos os diferenciavam.

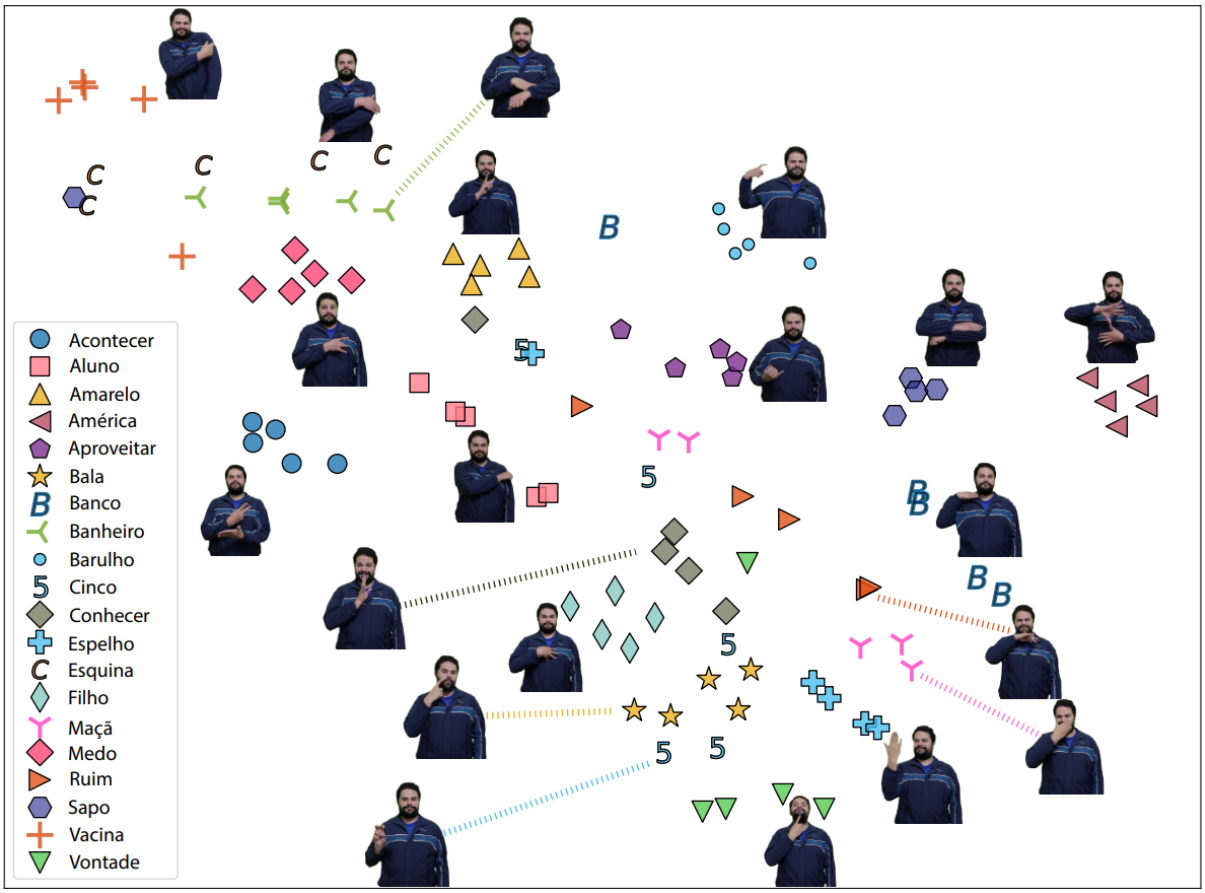


Figura 19 – Visualização dos sinais da MINDS-Libras utilizando a técnica t-SNE. Fonte: Castro (2020).

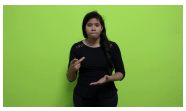
O número de vezes que cada sinal foi gravado é um parâmetro com grande variação nas bases de dados disponíveis na literatura. Há trabalhos que realizam um (Conly et al., 2013; Caselli et al., 2017), cinco (Almeida, 2014a; Ronchetti et al., 2016), dez (Li, 2017; Oszust e Wysocki, 2016a; Almeida et al., 2016) e até vinte (Oszust e Wysocki, 2016b) repetições por sinal. A escolha por capturar 5 repetições para cada gravação dos sinais da MINDS-Libras considerou (i) a diferença no tempo para execução para diferentes sinalizadores; (ii) o espaço físico necessário para o armazenamento desses dados digitais; e (iii) possíveis perdas durante o processo de captura, garantindo amostras de cada sinal/sinalizador. A priorização da repetição nas gravações em detrimento a quantidade de sinais é importante pois leva em consideração a variação entre as execuções de um mesmo sinalizador.

Por fim e entendendo a necessidade de complementar as informações relativas aos sinais gravados e facilitar a sua reprodução, o Apêndice B apresenta o significado e a definição dos parâmetros fonológicos de cada sinal que compõe a MINDS-Libras.

3.2.2 Sinalizadores

É desejável que uma base de sinais possua diferentes sinalizadores para minimizar possíveis vieses dos dados. Com esse cuidado, os sinais foram executados por 12 pessoas[10], variando-se o sexo, idade e o conhecimento na Língua Brasileira de Sinais, como ilustra a Tabela 4. Além disso, não houve padronização de vestimentas dos sinalizadores, pois espera-se que um algoritmo de classificação dos sinais seja robusto o suficiente para identificar as amostras independente da roupa do sinalizador. Cada sinalizador gravou 100[11] amostras, em um processo com duração aproximada de 2 horas.

Tabela 4 – Características dos sinalizadores.

Sinalizador	Sexo	Idade (anos)	Conhecimento prévio	Cor da Roupa
	Masculino	30-40	Fluente (surdo)	Azul
	Masculino	20-30	Fluente (intérprete)	Preta
	Feminino	20-30	Intermediário	Preta
	Feminino	30-40	Fluente (professora)	Vinho
	Feminino	30-40	Intermediário	Preta
	Feminino	30-40	Fluente (intérprete)	Branca
	Feminino	20-30	Básico	Preta
	Feminino	40-50	Básico	Preta

Continue na próxima página

[10] Os sinalizadores assinaram um termo para uso da imagem, em acordo com o projeto aprovado no edital 169/2015 do Instituto Federal de Minas Gerais, participando de forma voluntária.

[11] Número de gravações da cada sinalizador: 20 sinais × 5 repetições = 100 gravações.

Tabela 4 – *Continue na página anterior*

	Masculino	20-30	Intermediário	Preta
	Feminino	20-30	Intermediário	Preta
	Feminino	20-30	Intermediário	Preta
	Masculino	20-30	Intermediário	Branca

3.2.3 Sensores

Para capturar os vídeos dos sinais em Libras, foram utilizados uma câmera digital profissional Canon EOS Rebel t5i e o sensor RGB-D *Kinect v2* para *Xbox One*. O primeiro dispositivo é uma tecnologia que permite gravar vídeos em RGB e pode ser substituído por qualquer outra câmera fotográfica. A escolha pelo uso do sensor RGB-D se deu pela popularidade do *Kinect*, baixo custo e capacidade de gravar vídeos que contém informações de profundidade, pontos do corpo e da face. A Figura 20 ilustra essas características.

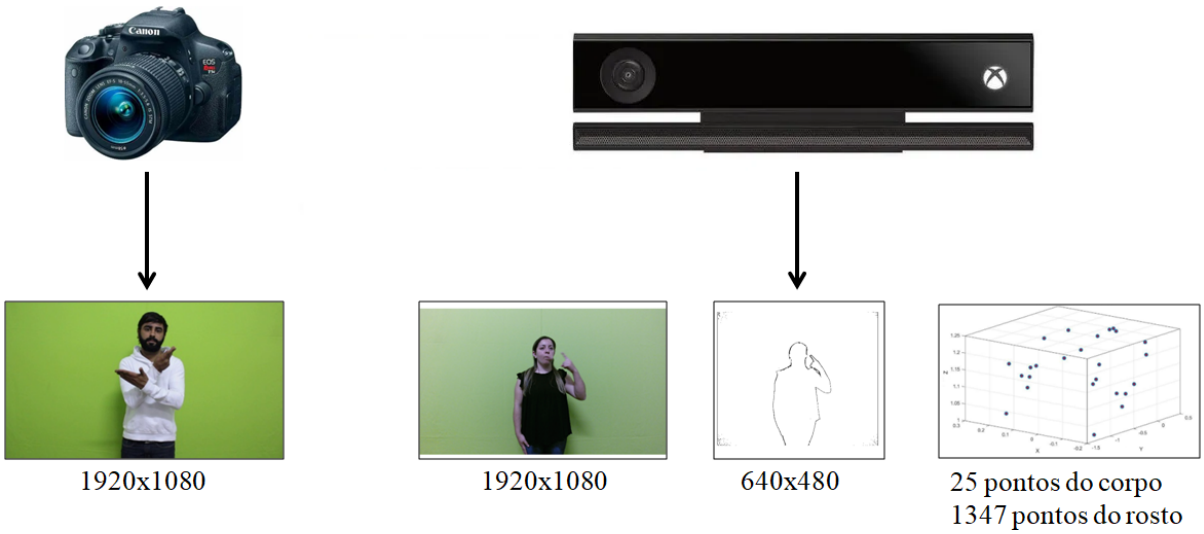


Figura 20 – Dados disponibilizados pelos dispositivos de captura: câmera RGB e sensor RGB-D.

A câmera grava vídeos em RGB com resolução 1920×1080 e formato ".mp4". Para reduzir o tempo de gravação, a câmera permaneceu ligada durante o registro das 5 repetições de cada sinal/sinalizador. Dessa forma, houve a necessidade de se realizar um pós-processamento para dividir as amostras por sinal. Nesse caso, cada amostra terá um número de quadros específico e não padronizado.

Outra questão importante está relacionada aos dados registrados pelo sensor RGB-D. Para obter de forma sincronizada todos os dados, a biblioteca Kin2 (Terven e Córdova-Esparza, 2016) do Matlab[12] foi utilizada. Como o sensor RGB-D precisou de um tempo para identificar corretamente todas as informações, uma interface foi implementada permitindo que a gravação iniciasse apenas por meio do comando do usuário. Em seguida, 5 segundos de execução de sinal foram registrados, totalizando 150 quadros (30 quadros por segundo) por amostra. Esse tempo de execução foi suficiente para capturar os sinais e foi um bom limiar para evitar o sistema de erro/travamento devido ao armazenamento de dados na memória.

O sensor RGB-D grava vídeos em RGB com resolução 1920×1080, vídeos da informação de profundidade em 640×480 *pixels*, dados de 25 juntas do corpo e de 1347 pontos da face do sinalizador. Todas essas informações estavam, inicialmente, em formato próprio do Matlab (".mat"). No entanto, para garantir que o uso desses dados não seja restrito a pesquisadores que utilizam o *software*, houve também um pós-processamento para disponibilizar os dados em formatos universais. Dessa forma, os vídeos foram convertidos para ".mp4" e os arquivos relativos aos pontos do corpo e da face em arquivo ".txt".

Os dispositivos aqui empregados apresentam características próprias e que tem atualizações implementadas constantemente. Com isso, as peculiaridades descritas podem não ser necessárias, dependendo da versão utilizada. O mais marcante no protocolo não é o equipamento em si, mas os formatos que os dados foram disponibilizados.

3.2.4 Cenário das Gravações

As gravações dos sinais ocorreram em um estúdio com iluminação controlada e com plano de fundo fixo feito de tecido *Chroma Key* com largura de 2,80 metros, como ilustra a Figura 21. O estúdio de gravação foi construído para permitir adicionar ou remover diferentes *backgrounds* nos vídeos de maneira a explorar o desempenho de algoritmos com base em padrões visuais. A distância entre o sinalizador e o sensor RGB-D foi fixada[13] em $\approx 1,60$ metros e em $\approx 2,00$ metros para a câmera RGB. O ângulo visto pelos sensores compreende o movimento dos membros superiores.

3.2.5 Dados Disponibilizados

No final do processo de gravação, a base MINDS-Libras possui, aproximadamente, 115GB de informação, distribuída em arquivos de vídeo e de texto. Os dados da câmera RGB e do sensor RGB-D foram disponibilizados separadamente e podem ser acessados

[12] Matlab versão *Student* 2018.

[13] Na base de dados MINDS-Libras as distâncias entre os sensores e o sinalizador foram fixas. Não houve normalização no tamanho das imagens. Para os casos em que a profundidade é desconhecida, pode-se aplicar a normalização pela largura dos ombros do sinalizador ou qualquer outra característica física deles.

Figura 21 – Cenário criado para a gravação dos sinais. Ambiente interno, com iluminação controlada.

em Minds (2019). Algumas informações da câmera RGB foram perdidas no processo de gravação devido a mesma ter entrado em modo de descanso e problemas na sua bateria. Optou-se por não regravar tais amostras e registrar os dados faltantes[14], pois eles estão presentes na gravação realizada pelo sensor RGB-D. Devido a essas perdas, das 1.200 amostras inicialmente planejadas, estão disponíveis 1.155 gravações[15]. As gravações dos sinais que não estão presentes na MINDS-Libras (câmera RGB) são referentes ao:

- Sinalizador 3: sinais "aluno", "América" e "cinco";
- Sinalizador 4: sinal "filho"; e
- Sinalizador 9: sinais "amarelo", "banheiro", "conhecer", "esquina" e "medo".

Os dados da câmera RGB somam 64, 8GB. O nome de cada arquivo ".mp4" foi configurado da seguinte forma: "*01AcontecerSinalizador03-2.mp4*" indicando o número do sinal[16] e o seu nome (Exemplo: 01Acontecer), o sinalizador e seu respectivo número[17] (Exemplo: Sinalizador03) e a gravação[18] correspondente (Exemplo: 2). O diretório referente a esses dados é ilustrado pela Figura 22.

As gravações realizadas pelo sensor RGB-D estão completas e totalizam 50, 1GB. Dessa forma tem-se 1200 amostras dos seguintes elementos: vídeos em RGB, vídeos em

[14] Veja a Tabela 7, Seção 3.3.1.
[15] Amostras disponíveis da câmera RGB: 1200 - (5 gravações × 9 sinais) = 1155 amostras.
[16] Número do sinal variando de 01 a 20 (dois dígitos).
[17] Número do sinalizador variando de 01 a 12 (dois dígitos).
[18] Gravação do sinal variando de 1 a 5.

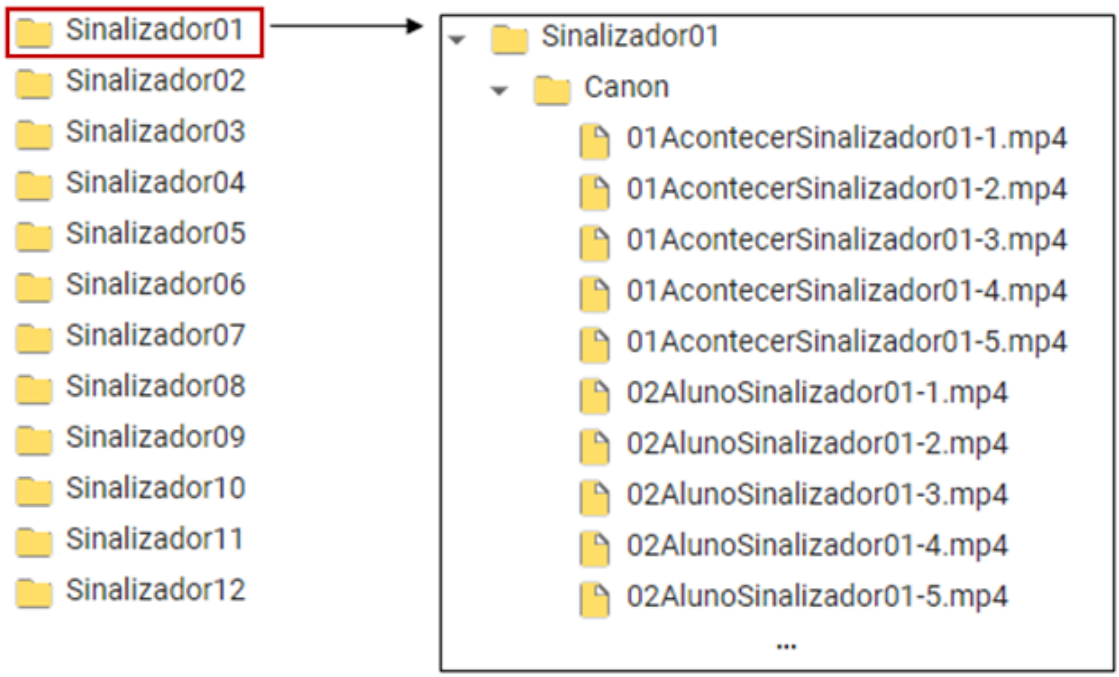


Figura 22 – Organização dos dados disponibilizados referentes à câmera RGB.

profundidade, um ".txt" com informação das 25 juntas do corpo e um com parâmetros de 1347 pontos da face. O nome dos arquivos também seguiram um padrão como, por exemplo, "*2-05Aproveitar_ 1Body.txt*" indicando o número do sinalizador[19] (Exemplo: 2), o número do sinal e o seu nome (Exemplo: 05Aproveitar), a gravação (Exemplo: 1), a informação contida naquele arquivo[20] (Exemplo: *Body*) e a extensão do arquivo[21] (Exemplo: ".txt"). A estrutura dos arquivos ".txt" com as informações registradas em cada um deles é descrita nas Tabelas 5 e 6. Para cada um, corpo e face, há 7 variáveis diferentes retornadas pelo sensor e organizadas sequencialmente nas linhas do arquivo. Por fim, a Figura 23 exemplifica a organização destes dados.

Tabela 5 – Arquivo do sensor RGB-D relativo aos dados do corpo.

Arquivo	Dado	Linha	# Coluna	Descrição
Corpo	*Position*	1	25	coordenada x de 25 pontos do corpo
		2	25	coordenada y de 25 pontos do corpo
		3	25	coordenada z de 25 pontos do corpo
	Orientation	4	25	coordenada x de 25 pontos do corpo dada em quatérnion
		5	25	coordenada y de 25 pontos do corpo dada em quatérnion
		6	25	coordenada z de 25 pontos do corpo dada em quatérnion
	TrackingState	7	25	estado de rastreamento de cada ponto do corpo
	LeftHandState	8	1	estado da mão esquerda
	RightHandState	9	1	estado da mão direita
	ColorPosition	10	25	posição da coordenada x no quadro RGB
		11	25	posição da coordenada y no quadro RGB
	DepthPosition	12	25	posição da coordenada x no quadro de profundidade
		13	25	posição da coordenada y no quadro de profundidade

[19] Neste caso, o número dos sinalizadores variaram de 1 a 12.
[20] *Body*, Face, RGB ou *Depth*.
[21] ".mp4" ou ".txt".

Tabela 6 – Arquivo do sensor RGB-D relativo aos dados da face.

Arquivo	Dado	Linha	# Coluna	Descrição
Face	*FaceBox*	1	4	vértices do retângulo em torno da face do sinalizador
	FaceRotation	2	3	orientação da face expressa em ângulos de Euler: *pitch*, *yaw*, *roll*
	HeadPivot	3	3	coordenadas x-y-z do ponto de referência da face
	AnimationUnits	4	17	17 unidades de animação restreadas do rosto
	FaceModel	5	1347	posição da coordenada x da face
		6	1347	posição da coordenada y da face
		7	1347	posição da coordenada z da face
	ColorFaceModel	8	1347	posição da coordenada x no quadro RGB
		9	1347	posição da coordenada y no quadro RGB
	DepthFaceModel	10	1347	posição da coordenada x no quadro de profundidade
		11	1347	posição da coordenada y no quadro de profundidade

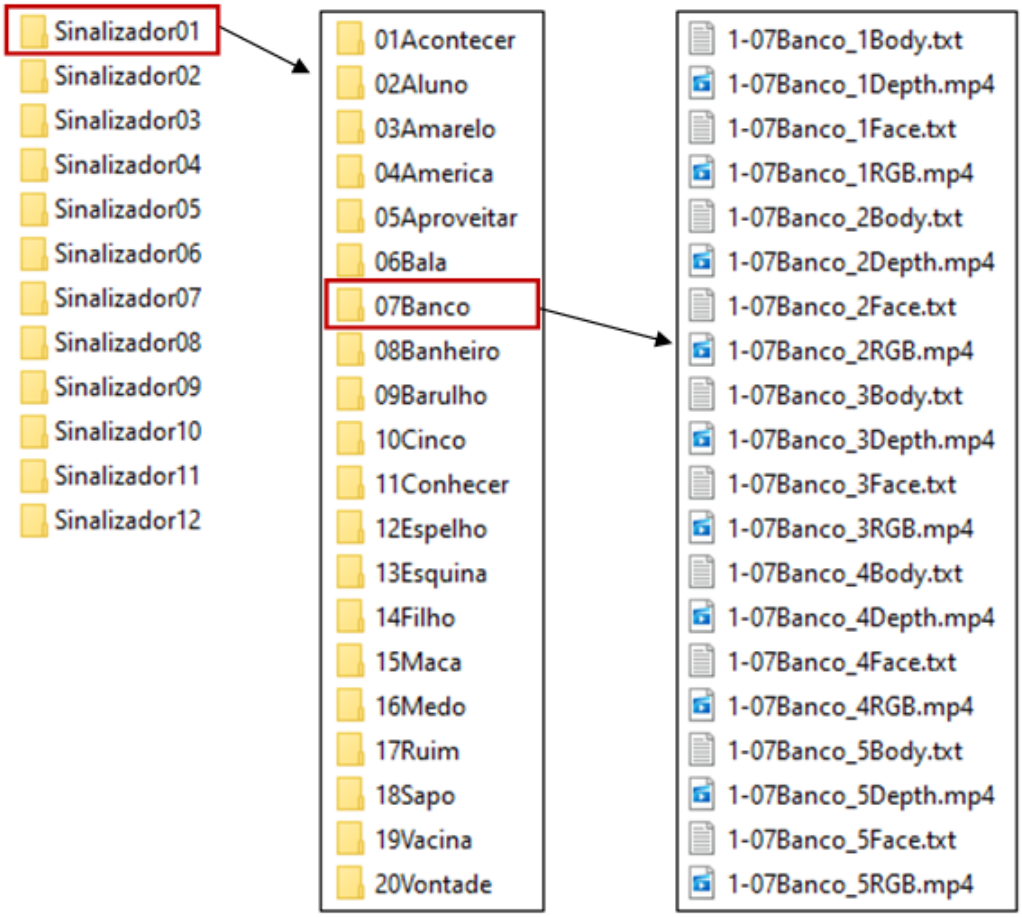


Figura 23 – Organização dos dados disponibilizados referente ao sensor RGB-D.

A utilização de diferentes sensores teve como intuito mostrar que vários são os dispositivos que permitem a aquisição de dados para a criação de uma base de dados. A fonte não influencia nos resultados.

3.3 Análise Descritiva

A base de dados foi criada após um processo de escolha criteriosa dos parâmetros que a compõem (sinais, sinalizadores, sensores, ambiente de gravação), juntamente com

a definição de um protocolo de gravação. Realizar a análise descritiva da MINDS-Libras finaliza e completa o processo de criação da base, pois permite descrever com detalhes seus aspectos e entender o comportamento dos seus elementos. Além disso, em geral, é difícil entender o formato dos dados nas bases disponíveis publicamente, pois não são bem documentados. Dessa forma, o estudo realizado nesta seção, disponível em Rezende (2020), apresenta as características relevantes dos dados visando a sua aplicação em sistemas de reconhecimento automático de sinais em Libras e de gestos.

3.3.1 Dados da câmera RGB: vídeos

Como apresentado na Seção 3.2.3, dois sensores foram utilizados para a captura dos sinais: uma câmera RGB e um sensor RGB-D. Em relação ao primeiro, cada amostra é disponibilizada em vídeo, com extensão ".mp4", ilustrando como o sinal é executado. Uma característica desses vídeos é que cada um deles é composto por um número diferente de quadros/imagens, devido à prosódia[22] do sinal. Apesar desse termo ser oriundo das línguas orais, Goes (2019) declara que ele se aplica às línguas de sinais, sendo caracterizado pela duração do sinal, pelas pausas, expressões corporais e faciais, isto é, atributos que auxiliam no significado de um sinal.

Das amostras que formam a base MINDS-Libras, o número de quadros que compõem os vídeos varia de 71 a 233, como mostra a Tabela 7. Analisando esses valores, percebeu-se que para um mesmo sinalizador, há uma similaridade em relação ao número de quadros de suas amostras, ou seja, a velocidade com que ele executa os sinais é muito semelhante, mesmo em se tratando de sinais diferentes. Além disso, a relação do número de quadros, para um mesmo sinal, varia quando muda o sinalizador. A justificativa para esse caso se mantém e está diretamente ligada à prosódia, isto é, cada sinalizador possui uma forma própria de executar os sinais que interfere diretamente no tempo de duração das amostras. Por fim, a Tabela 7 destaca os dados faltantes, como abordado na Seção 3.2.5.

Analisar essa variação é relevante quando se trabalha com algoritmos de classificação que requerem como entrada vetores de características de mesma dimensão e, também, quando se tem limitação em relação ao poder de processamento dos dados. Nesse último ponto vale ressaltar que o tratamento de vídeos possui um alto custo computacional e, muitas das vezes, é necessário realizar um pós-processamento nos dados para minimizar esse consumo.

[22] Prosódia é o estudo do ritmo, entonação e demais atributos correlatos na fala. Ela descreve todas as propriedades acústicas da fala que não podem ser preditas pela transcrição ortográfica (ou similar).

Tabela 7 – Número de quadros referente aos vídeos da câmera RGB para os sinais: (01) acontecer, (02) aluno, (03) amarelo, (04) América, (05) aproveitar, (06) bala, (07) banco, (08) banheiro, (09) barulho, (10) cinco, (11) conhecer, (12) espelho, (13) esquina, (14) filho, (15) maçã, (16) medo, (17) ruim, (18) sapo, (19) vacina e (20) vontade.

		Sinal																			
Sinalizador	**Gravação**	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
01	1	104	130	164	111	113	127	141	113	130	116	114	200	134	137	116	112	135	113	155	154
	2	130	143	183	148	116	164	148	150	196	150	125	145	113	125	109	143	124	137	112	148
	3	120	150	139	151	118	122	129	201	168	157	125	145	112	136	141	141	145	130	134	141
	4	121	155	162	154	112	125	173	148	160	130	118	116	120	133	125	157	127	145	101	136
	5	131	169	145	153	111	137	141	144	166	152	126	140	115	173	141	150	141	132	156	146
02	1	104	157	125	160	141	142	122	150	125	123	128	148	165	118	124	128	129	198	139	133
	2	123	132	167	158	91	117	180	199	122	155	171	139	147	123	162	173	135	147	130	143
	3	107	157	155	153	99	127	125	129	132	142	156	169	199	121	169	148	168	168	180	150
	4	144	164	164	169	143	121	136	150	128	141	132	168	154	127	167	158	166	165	150	200
	5	110	122	104	141	178	124	157	152	128	150	127	178	150	135	158	127	166	142	183	166
03	1	141	-	110	-	125	146	126	121	138	-	131	111	134	110	121	113	124	100	144	144
	2	118	-	139	-	103	183	125	107	134	-	114	187	152	118	143	141	130	131	161	138
	3	164	-	148	-	139	123	125	120	129	-	100	122	125	142	139	165	182	123	152	160
	4	126	-	128	-	156	131	139	145	160	-	95	168	147	112	139	151	119	145	166	148
	5	156	-	148	-	100	107	129	131	151	-	131	155	184	117	117	124	113	160	191	114
04	1	149	157	168	221	138	174	149	182	180	185	162	162	189	-	191	186	176	184	153	161
	2	152	160	173	174	136	184	150	168	221	216	163	138	166	-	225	145	152	215	174	163
	3	155	161	129	145	169	168	180	176	196	163	159	180	179	-	193	180	167	200	205	199

Continue na próxima página

Tabela 7 – *Continue na página anterior*

	4	140	167	183	196	146	168	159	162	185	153	168	173	179	-	150	182	147	191	185	176
	5	171	189	174	212	139	160	166	158	161	181	184	214	187	-	175	161	161	185	182	179
05	1	129	168	160	114	114	136	136	132	126	139	109	121	146	127	111	111	137	175	159	127
	2	108	173	143	152	88	125	142	118	139	118	149	126	145	135	155	144	128	175	124	122
	3	120	159	101	139	109	97	112	151	154	123	106	163	143	155	152	118	116	169	127	132
	4	125	171	98	155	109	137	132	127	164	101	145	119	149	100	159	97	111	144	121	137
	5	97	162	118	139	125	135	141	120	134	128	116	120	158	139	127	139	108	146	145	136
06	1	107	169	155	156	125	144	136	145	139	115	140	130	166	121	159	119	133	146	134	166
	2	146	187	134	153	109	175	144	173	145	118	160	141	170	114	154	135	126	157	139	132
	3	152	173	131	138	123	136	152	162	134	119	140	147	170	107	164	142	112	166	145	153
	4	121	191	134	139	89	134	175	178	156	106	155	148	203	116	164	136	121	138	145	139
	5	103	174	150	143	106	145	148	159	127	102	164	152	155	121	161	127	126	145	144	154
07	1	142	117	157	137	109	118	109	117	136	113	123	113	146	124	116	121	116	152	145	130
	2	121	103	159	147	96	138	132	118	184	143	133	108	159	102	148	107	125	133	166	139
	3	136	122	146	144	109	107	125	124	155	148	111	152	131	100	126	120	129	162	146	116
	4	127	134	122	130	107	104	128	138	148	113	113	130	121	136	145	130	120	168	156	130
	5	116	117	132	139	93	131	133	139	132	135	143	121	134	111	129	125	129	178	123	120
08	1	81	125	109	105	71	118	102	112	121	93	112	130	154	102	133	121	132	115	95	136
	2	97	122	102	126	79	107	105	107	150	123	129	105	110	100	100	111	139	128	103	178
	3	81	122	112	130	78	91	96	106	135	131	120	118	116	109	132	107	142	158	98	162
	4	93	159	96	140	82	128	104	105	169	107	109	100	125	102	123	107	114	130	118	153
	5	105	139	110	126	101	123	93	125	158	136	108	105	123	90	120	130	135	135	122	137
09	1	119	130	-	191	153	159	127	-	143	130	-	125	-	155	123	-	137	165	154	160
	2	151	132	-	146	131	126	156	-	162	120	-	128	-	142	114	-	127	140	156	163
	3	126	171	-	146	133	148	149	-	142	165	-	141	-	120	149	-	116	169	194	144

Continue na próxima página

Tabela 7 – *Continue na página anterior*

	4	126	126	-	158	148	125	135	-	157	153	-	151	-	137	144	-	151	171	196	133
	5	121	160	-	155	103	145	140	-	159	107	-	131	-	134	121	-	137	137	170	146
10	1	95	102	107	102	93	97	88	87	141	113	95	93	105	89	106	92	89	130	100	91
	2	93	101	145	128	88	113	101	107	121	136	91	114	123	96	119	115	115	111	107	93
	3	129	114	125	102	103	104	95	104	137	123	100	137	136	77	113	99	91	106	91	104
	4	98	106	137	106	88	107	89	87	145	117	116	130	102	80	111	102	87	109	91	100
	5	100	109	116	116	89	98	105	93	136	141	94	145	133	77	122	88	145	123	102	102
11	1	136	147	169	128	121	165	144	141	152	136	157	131	138	141	134	143	157	127	129	155
	2	124	155	189	154	135	157	160	143	182	148	130	127	126	139	164	131	146	148	143	155
	3	146	160	177	131	123	150	158	168	166	157	168	167	127	155	165	136	142	139	166	146
	4	146	169	200	147	129	164	150	136	160	148	154	168	142	138	155	125	146	164	146	168
	5	131	141	169	137	117	167	156	141	164	150	145	141	143	150	155	122	137	157	136	159
12	1	138	233	207	178	132	168	157	210	153	163	166	151	136	161	153	125	126	186	141	158
	2	123	201	158	168	113	178	191	186	154	137	179	201	178	193	159	155	177	195	130	143
	3	157	182	192	184	123	211	152	173	175	143	163	194	203	165	155	165	158	172	150	174
	4	146	179	155	196	111	183	180	166	141	122	181	193	170	162	170	126	149	194	148	132
	5	138	208	169	185	101	185	138	157	176	136	169	198	157	172	184	150	155	171	139	167

3.3.2 Dados do sensor RGB-D: juntas corpo

Em relação ao sensor RGB-D foram disponibilizados: (i) vídeos dos sinais em formato "mp4", arquivos em formato ".txt" contendo sete variáveis referentes à (ii) 25 juntas do corpo e (iii) à 1347 pontos da face, estruturados de acordo com a descrição apresentada nas Tabelas 5 e 6 (Seção 3.2.5).

No que se refere aos dados do corpo, o sensor retorna as seguintes informações:

1. *Position*: coordenadas x-y-z de 25 juntas/pontos/articulações do corpo distribuídas espacialmente de acordo com a Figura 24. Seus valores, dados em metros, tem como referência o centro geométrico do sensor e variam entre $[-2.2, +2.2]$ para x, $[-1.6, +1.6]$ para y e $[0.0, 4.0]$ para z. Do ponto de vista do sensor, x cresce à esquerda, y para cima e z na direção em que o sensor está voltado, como ilustra a Figura 25. Como as gravações capturaram as informações da parte superior do corpo, sugere-se excluir os pontos 13 a 20 relativos aos membros inferiores do esqueleto. Dessa forma, há 17 pontos que representam as informações significativas do sinal, conforme apresentado nas Figuras 26b e 26c.

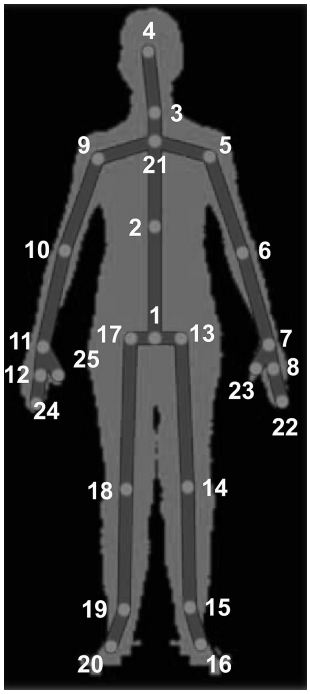


Figura 24 – Posição das 25 juntas capturadas pelo sensor RGB-D: 1−base da coluna, 2−coluna, 3−pescoço, 4−cabeça, 5-ombro esquerdo, 6-cotovelo esquerdo, 7-pulso esquerdo, 8-mão esquerda, 9-ombro direito, 10-cotovelo direito, 11-pulso direito, 12-mão direita, 13-quadril à esquerda, 14-joelho esquerdo, 15-tornozelo esquerdo, 16-pé esquerdo, 17-quadril à direita, 18-joelho direito, 19-tornozelo direito, 20-pé direito, 21-ombro/coluna, 22-ponta da mão esquerda, 23-polegar esquerdo, 24-ponta da mão direita e 25-polegar direito.
Adaptado de: Patrícia Rocha et al. (2018).

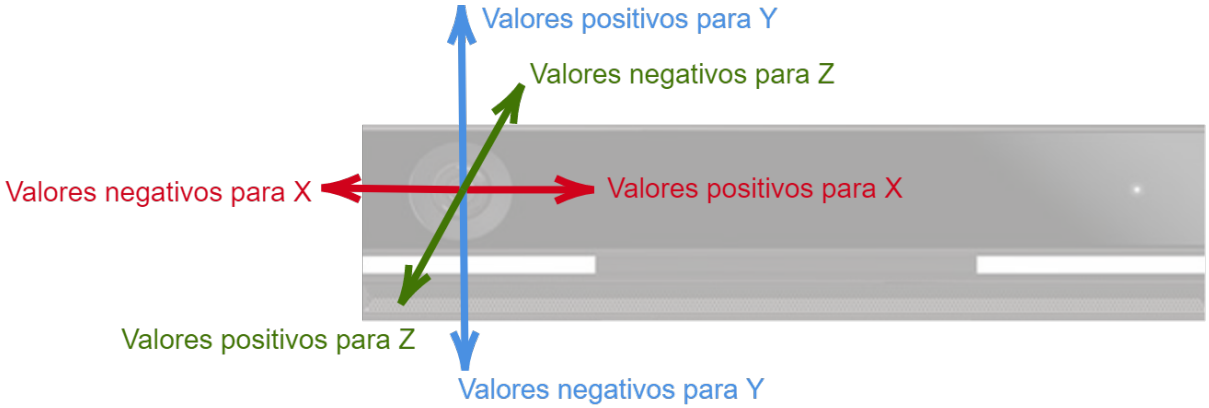
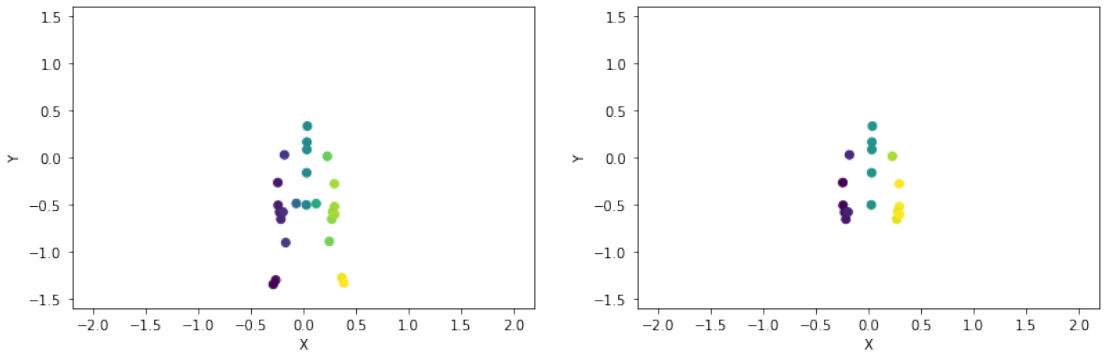
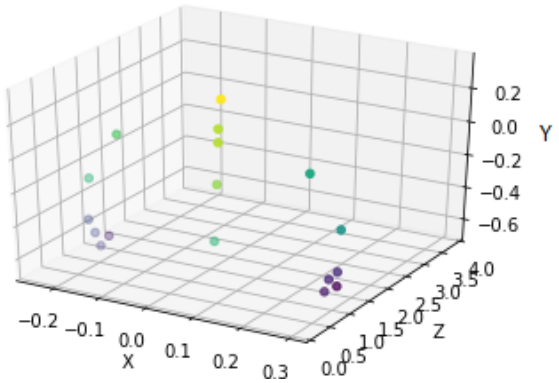


Figura 25 – Referência dos eixos com base ao centro geométrico do sensor RGB-D.



(a) x-y dos 25 pontos do corpo

(b) x-y dos pontos dos membros superiores



(c) x-y-z dos pontos dos membros superiores

Figura 26 – Posição dos pontos do corpo para um quadro de uma amostra para x $\in$ $[-2.2, 2.2]$, y $\in$ $[-1.6, 1.6]$ e z $\in$ $[0, 4]$.

2. *Orientation*: fornece os ângulos de Euler, *pitch* (rotação em torno de x), *yaw* (rotação em torno de y) e *roll* (rotação em torno de z), que são utilizados para demonstrar o movimento realizado após uma rotação de cada um eixos. Esta variável não foi utilizada neste estudo, mas pode ser útil em cenários de animação. Vale ressaltar que há algumas bibliografias, como Terven e Córdova-Esparza (2016), que indicam

que esta variável pode ser disponibilizada em forma de quatérnios, dependendo do que for setado na configuração do software[23].

3. *TrackingState*: é um valor que indica o estado de rastreamento da junta, sendo rastreado = 2, inferido = 1 e não rastreado = 0, como ilustra a Figura 27. Para o último valor, a junta é ignorada. Analisando essa variável, percebeu-se que os pontos dos membros inferiores do corpo recebem *TrackingState* = 1, afirmando que os pontos foram estimados. Este valor corrobora com a análise realizada quando foi caracterizada a variável *Position*, em que se sugeriu que fossem excluídos os pontos 13 a 20. Vale ressaltar que os únicos pontos que foram rastreados corretamente em algumas amostras foram os pontos relativos ao quadril, mas que na lógica de movimento estabelecida, podem ser excluídos também.

Figura 27 – *TrackingState* de um quadro.

4. *LeftHandState* e *RightHandState*: o sensor RGB-D utilizado atribui 5 estados para a mão, sendo 0 - desconhecido, 1 - não monitorada, 2 - aberta, 3 - fechada e 4 - nem aberta e nem fechada, como ilustram as Figuras 28 e 29.

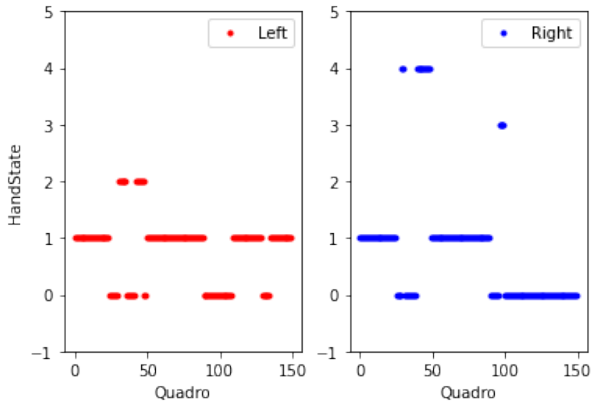


Figura 28 – Estados de cada uma das mãos ao londo dos quadros de uma amostra.

[23] Veja: https://medium.com/@lisajamhoury/understanding-kinect-v2-joints-and-coordinate-system-4f4b90b9df16 e https://github.com/jrterven/Kin2/blob/master/Mex/Kin2.m.

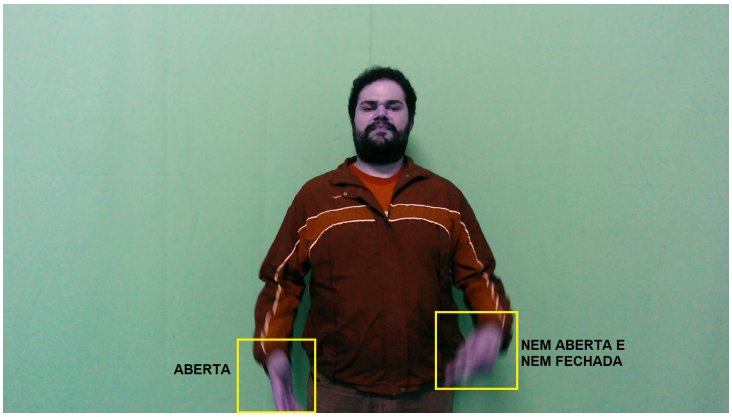
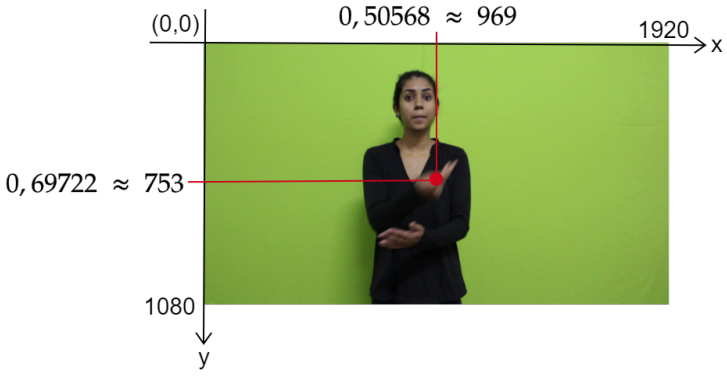
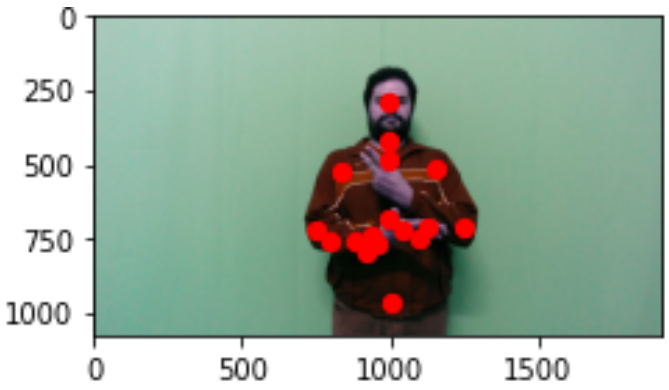


Figura 29 – *HandState* das mãos referente ao quadro 30 apresentado na Figura 28.

5. *ColorPosition*: a resolução da câmera RGB deste sensor é 1920×1080, tendo como referência o canto superior à esquerda da imagem, como mostra a Figura 30a. Dessa forma, os valores retornados pela variável *ColorPosition* correspondem aos mesmos x-y da posição, mas estão na escala de porcentagem, ou seja, $0,50468$ e $0,69722$ correspondem, na imagem, aos *pixels* 969 ($\approx 0,50468 \times 1920$) e 753 ($\approx 0,69722 \times 1080$), respectivamente. A Figura 30b apresenta os 17 pontos referentes aos membros superiores plotados no quadro em RGB.



(a) Referência



(b) *ColorPosition*

Figura 30 – 17 pontos do corpo plotados no quadro em RGB.

6. *DepthPosition*: utiliza a mesma regra aplicada para a variável *ColorPosition*. A diferença é que para este dado a resolução do sensor é 640×480 e o posicionamento das 25 juntas é adaptado para esta escala. A Figura 31 ilustra o exemplo de um quadro de profundidade com os 17 pontos do corpo.

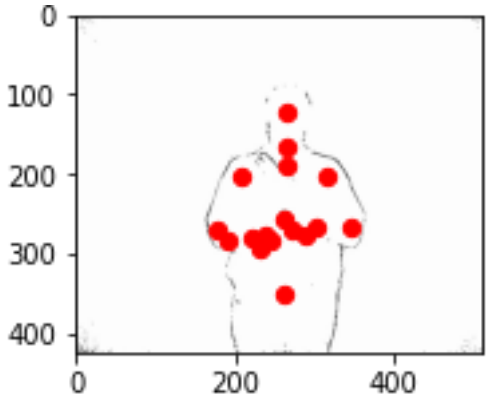


Figura 31 – Posição x-y dos 17 pontos do corpo plotados no quadro em profundidade.

Das sete variáveis disponibilizadas pelo sensor RGB-D, a *Position* é a que será utilizada para estudar os dados relativos ao corpo porque é a que trata da trajetória realizada durante a execução do sinal. Durante a análise dessa variável e da *TrackingState* foi abordada a exclusão de 8 pontos (Pontos 13 ao 20), pelo fato deles não terem sido captados durante a gravação dos sinais e os seus valores foram estimados pelo sensor. Aprofundando um pouco mais nesse estudo, uma análise gráfica das demais trajetórias foi realizada para verificar se os 17 pontos restantes possuíam informação significativa do sinal. A Figura 32 apresenta a variação das coordenadas x-y-z ao longo do vídeo para uma amostra da base de dados MINDS-Libras.

Os pontos relativos à coluna (Pontos 1 e 2), pescoço (Ponto 3), cabeça (Ponto 4) e ombros (Pontos 5, 9 e 21) permaneceram constantes durante toda a execução do sinal. Essa invariabilidade era esperada pelo fato do sinalizador possuir uma posição fixa e a movimentação corporal não ser um parâmetro representativo para a formação dos sinais. Mesmo mudando o sinal e o sinalizador esse padrão é mantido (falta de movimento) como ilustram, respectivamente, as Figuras 33 e 34. As pequenas oscilações ao longo do tempo vistas na Figura 33 são referentes à movimentos corporais insignificantes ou ruídos do sensor. Já as variações nas coordenadas apresentadas na Figura 34 são devido à amplitude do movimento realizado pelos sinalizadores. Dessa forma, sugere-se excluir, também, os pontos 1 a 5, 9 e 21.

Com isso, este estudo sugere que, se tratando dos juntas do corpo, 10 pontos podem ser utilizados para reconhecimento automático da MINDS-Libras. Estes pontos, exemplificados nas Figuras 35 e 36, são relativos à trajetória das mãos: cotovelos (Pontos

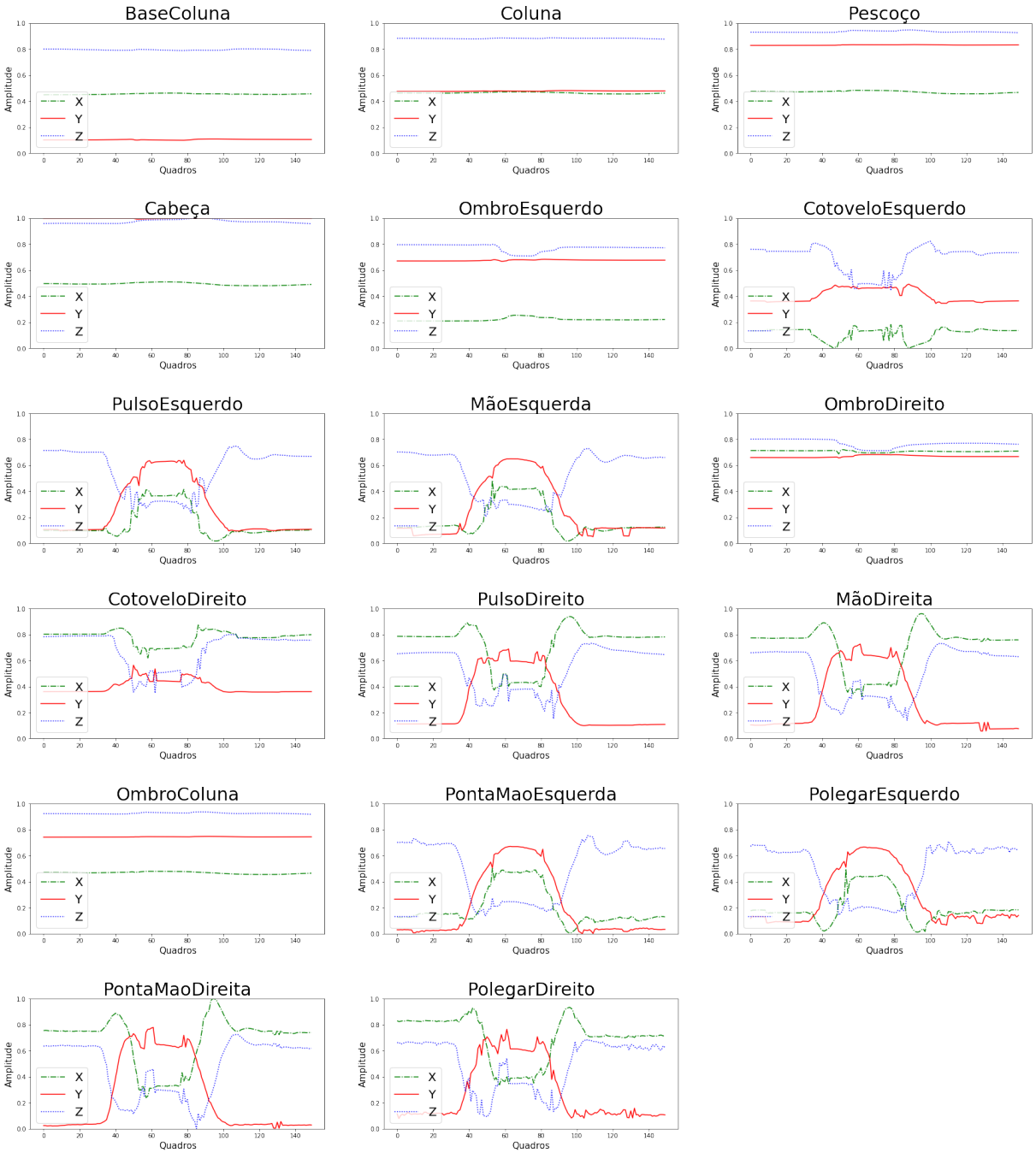


Figura 32 – Variação das coordenadas x (verde '-.'), y (vermelho '-') e z (azul '.') normalizadas para cada um dos 17 pontos do corpo, ao longo dos 150 quadros da amostra *1-01Acontecer_1Body.txt*.

6 e 10), pulsos (Pontos 7 e 11), mãos (Pontos 8 e 12), pontas das mãos (Pontos 22 e 24) e polegares (Pontos 23 e 25).

As Figuras 35 e 36 mostram que as curvas possuem um padrão de movimento quando se trata do mesmo lado de referência. Os cotovelos são os únicos pontos que possuem uma discrepância de trajetória em relação aos demais, devido a essa ser a junta que direciona o movimento das mãos e também funciona como uma junta de rotação do braço. O padrão das curvas para cada ponto se mantém quando se analisa as cinco

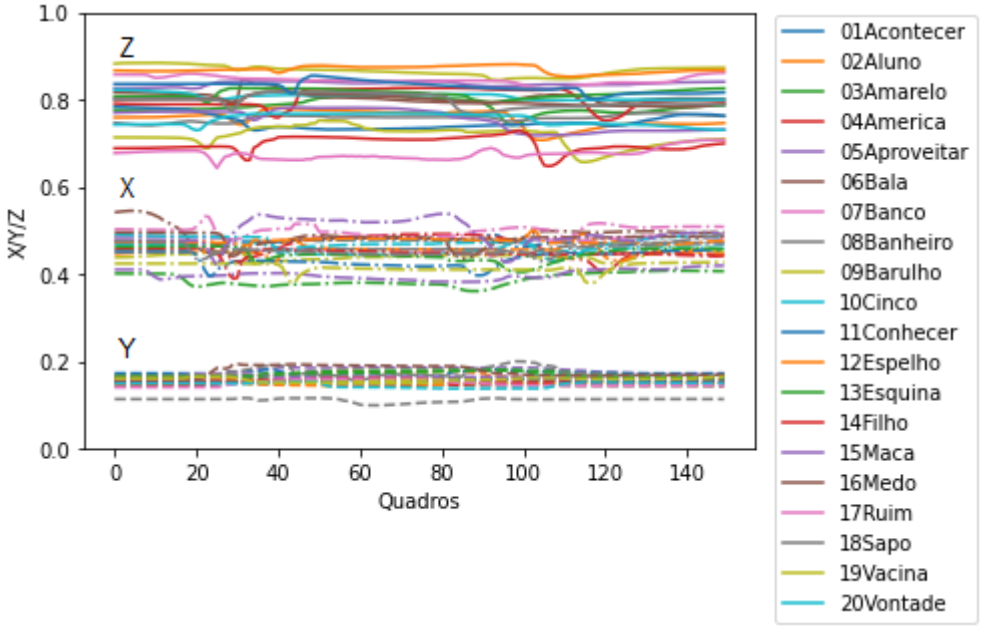


Figura 33 – Variação das coordenadas x ('-.'), y ('–') e z ('-') normalizadas do ponto 1 (base da coluna) para os 20 sinais da MINDS-Libras (sinalizador 1, gravação 1).

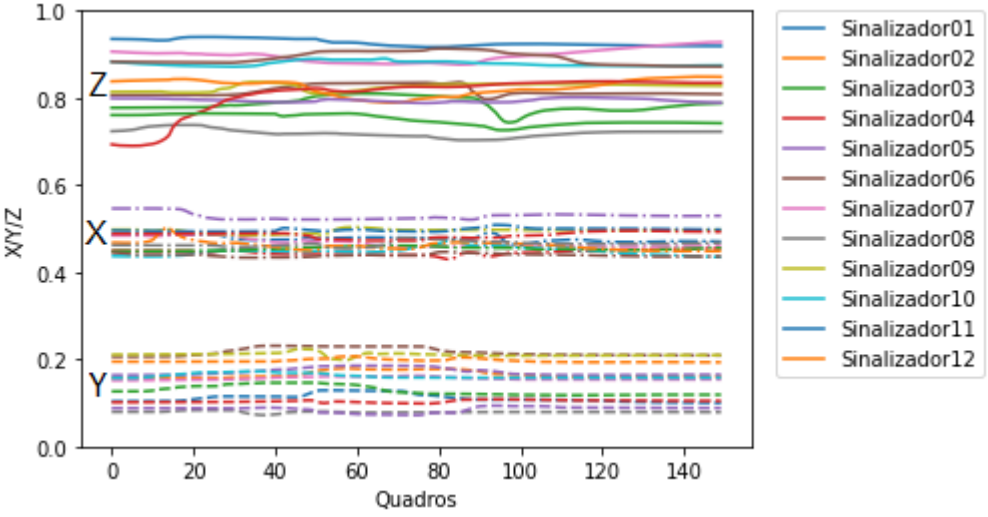


Figura 34 – Variação das coordenadas x ('-.'), y ('–') e z ('-') normalizadas do ponto 1 (base da coluna) para cada um dos sinalizadores da MINDS-Libras (Sinal "acontecer", gravação 1).

gravações de um mesmo sinal/sinalizador, como ilustra a Figura 37. Na mudança de sinalizador, a Figura 34 também apresentou essa característica. Entretanto, quando há movimentação do ponto em questão, há uma diferença na velocidade de execução de cada sinalizador atribuída à prosódia do sinal, como foi abordado na Seção 3.3.1 e ilustrado pela Figura 38. Sinalizadores diferentes vão gerar curvas que começam em tempos diferentes, com durações e amplitudes distintas. Dessa forma, para um determinado quadro do vídeo, provavelmente a execução do sinal não será coincidente para todos os sinalizadores.

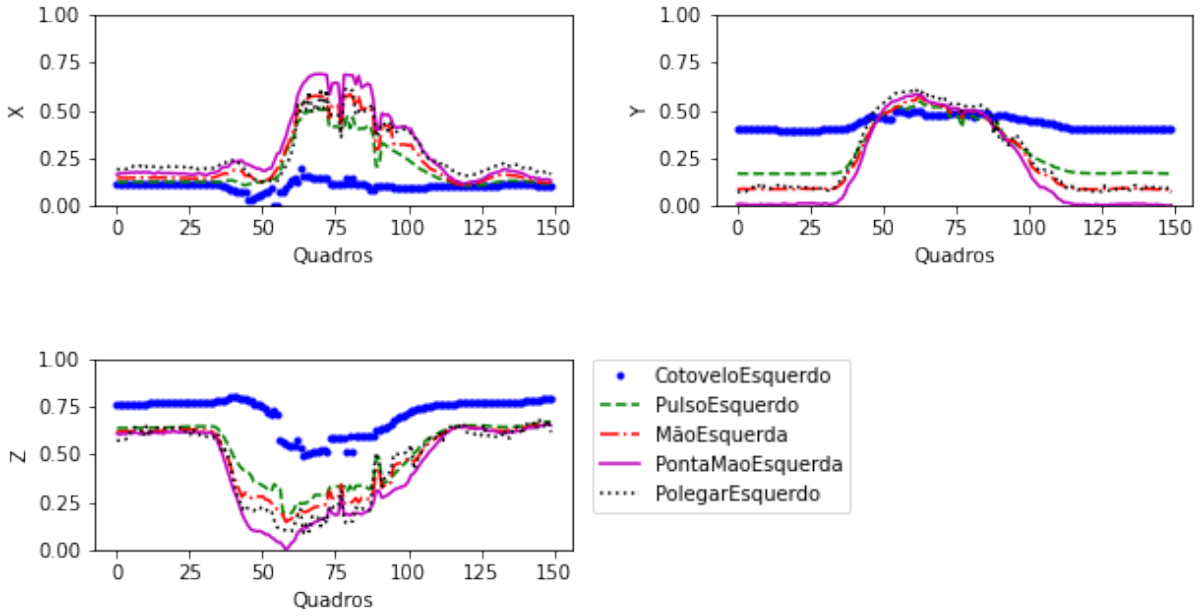


Figura 35 – Variação das coordenadas x, y e z normalizadas dos 5 pontos relativos à mão esquerda, ao longo dos 150 quadros da amostra *1-01Acontecer_1Body.txt*.

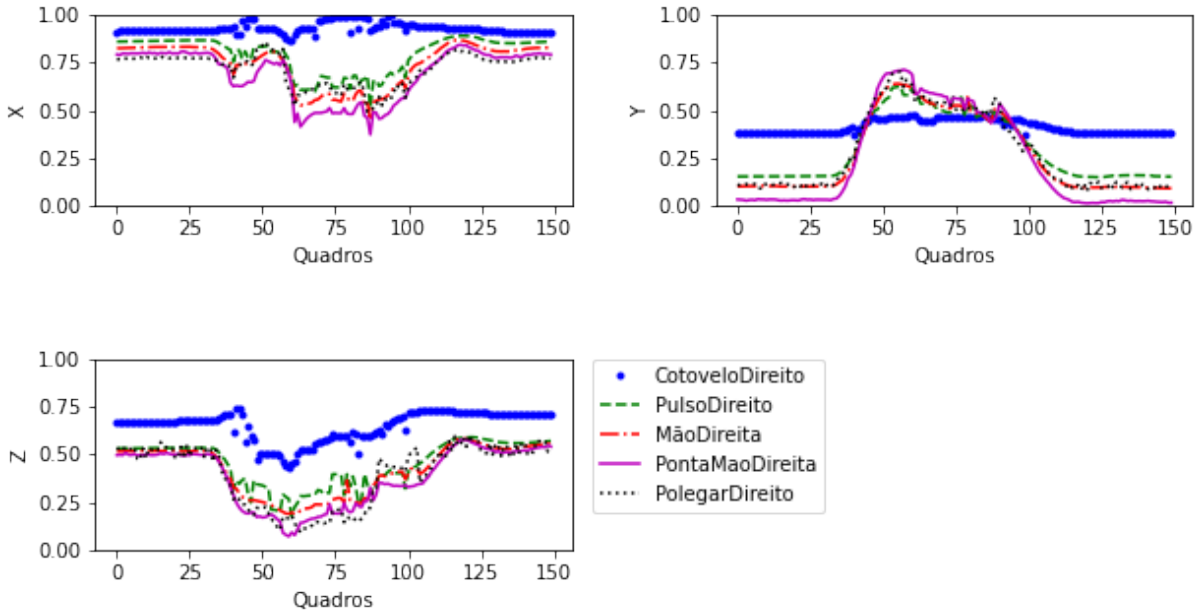


Figura 36 – Variação das coordenadas x, y e z normalizadas dos 5 pontos relativos à mão direita, ao longo dos 150 quadros da amostra *1-01Acontecer_1Body.txt*.

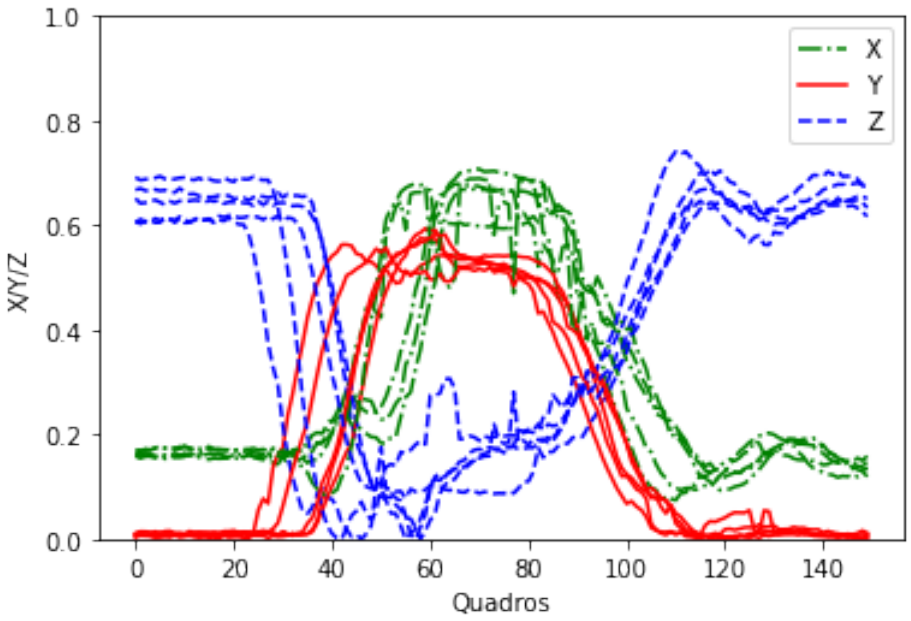


Figura 37 – Variação das coordenadas x (verde '-.'), y (vermelho '-') e z (azul '–') normalizadas do ponto 22-PontaMaoEsquerda para as cinco gravações do sinal "acontecer", sinalizador 01.

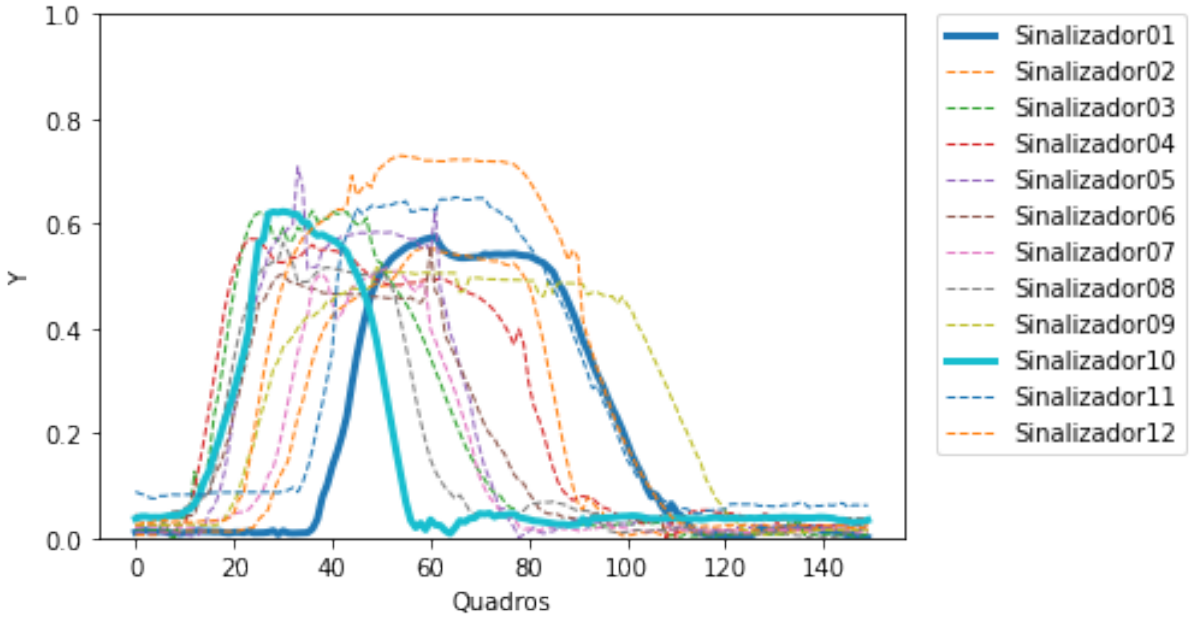


Figura 38 – Variação das coordenadas x normalizadas do ponto 22-PontaMaoEsquerda para a gravação 1 de cada sinalizador (Sinal "acontecer").

Muitas são as abordagens que permitem utilizar os pontos, sendo que o foco dado nesta seção foi a trajetória/movimento das mãos. Pode-se explorar a utilização de apenas um ponto para representar toda a mão, reduzindo a quantidade de informação a ser processada, analisá-los como uma série variando no tempo, ou até mesmo mesclar essas características com os quadros em RGB, por exemplo. Independente das informações que serão utilizadas no sistema de reconhecimento automático, a análise das juntas do corpo disponibilizadas pelo sensor RGB-D destacou que os pontos das mãos representam a informação de movimento que é relevante para identificar os sinais.

3.3.3 Dados do sensor RGB-D: pontos da face

Em relação aos dados da face, o sensor RGB-D retorna os seguintes parâmetros:

1. *FaceBox*: é a variável que permite desenhar um retângulo, limitando a face do sinalizador, como ilustra Figura 39. As coordenadas seguem a sequência x_1, y_1, x_2 e y_2, e são dadas em *pixels*, tendo como referência o canto superior esquerdo do quadro RGB, como apresentado na Figura 30a. Essa informação pode auxiliar os trabalhos que buscam utilizar os dados da expressão facial.

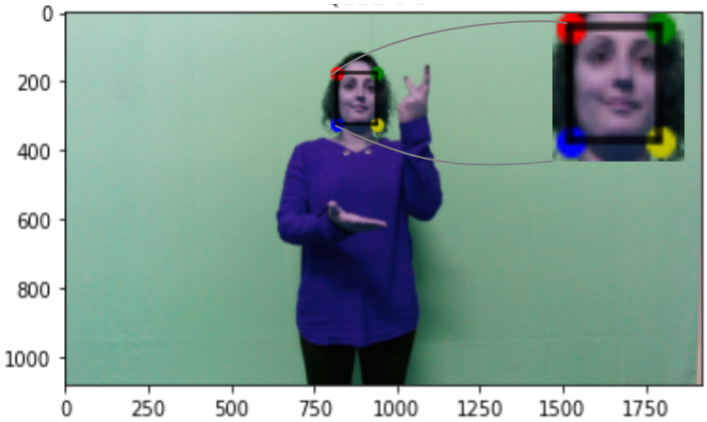


Figura 39 – *FaceBox*: coordenadas do retângulo que delimita a face do sinalizador, plotado no quadro RGB correspondente (Sinalizador 4).

2. *FaceRotation*: relativo à orientação da face e expressa como ângulos de Euler: *pitch* (rotação em torno de x), *yaw* (rotação em torno de y) e *roll* (rotação em torno de z), ilustrados na Figura 40. Como abordado na variável *Orientation* (dados do corpo), esses valores demonstram rotação do movimento e, consequentemente, podem ser utilizados em aplicações que trabalham com animação.

3. *HeadPivot*: indica as coordenadas x-y-z que são referência para o movimento da cabeça. Seus valores são dados em metros e tem como origem o centro geométrico do sensor (Figura 25). O *HeadPivot* pode ser uma variável utilizada para normalizar os demais dados, sendo uma referência para toda a base de dados.

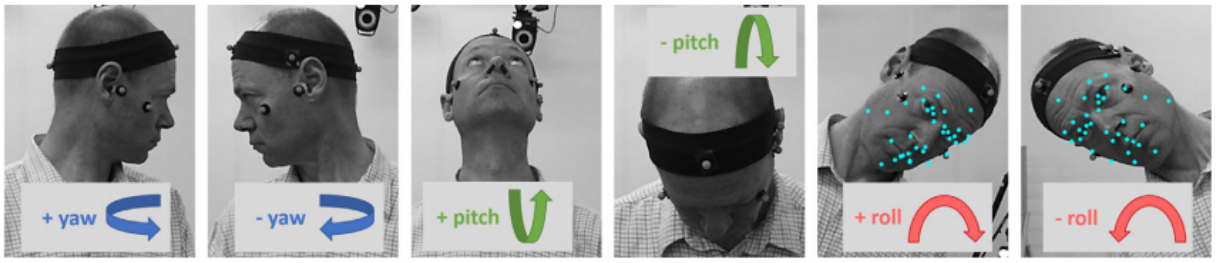
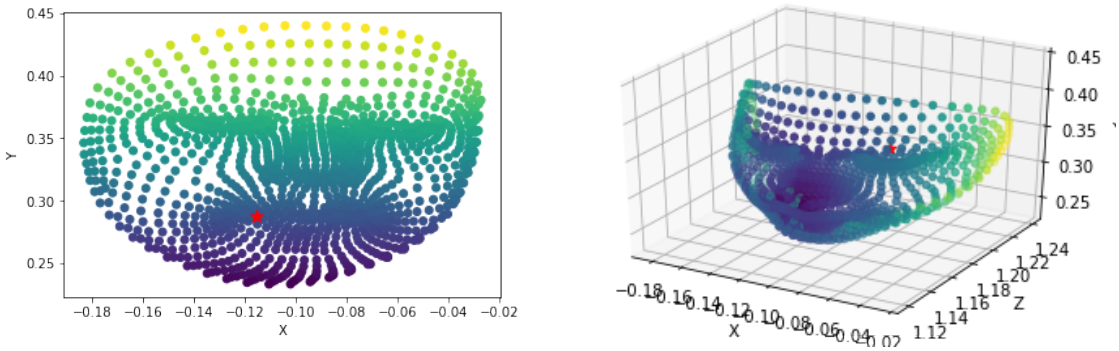


Figura 40 – Exemplos e orientação da cabeça.
Fonte: Darby et al. (2016).

4. *AnimationUnit*: apresenta 17 unidades de animação[24] para rastreamento de gestos faciais. Seus valores variam entre 0 e 1, indicando estados do rosto, isto é, se o olho está fechado ou a sobrancelha está mais baixa, por exemplo.

5. *FaceModel*: fornece as coordenadas x-y-z de 1347 pontos da face. Esta variável possui as mesmas características de *Position*: possui como referência o centro geométrico do sensor (Figura 25) e seus valores se encontram entre $[-2.2m, +2.2m]$ para x, $[-1.6m, +1.6m]$ para y e $[0.0m, 4.0m]$ para z. A Figura 41 ilustra a disposição desses pontos nos eixos e o *HeadPivot* (vermelho - '*') do quadro em questão.



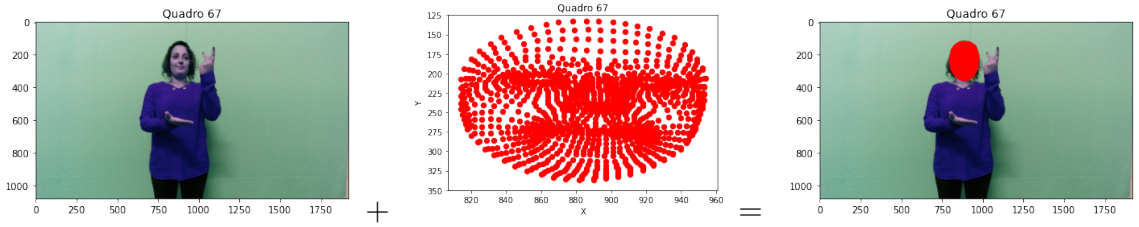
(a) Coordenadas x-y

(b) Coordenadas x-y-z

Figura 41 – *FaceModel*: coordenadas dos 1347 pontos relativos à face do sinalizador e *HeadPivot*: ponto de referência para o movimento da cabeça destacado em vermelho (*) (Sinalizador 4).

6. *ColorFaceModel*: os valores retornados pela variável *ColorFaceModel* correspondem aos mesmos x-y da *FaceModel*, mas estão na escala de porcentagem, com referência aos quadros em RGB (similar à variável *ColorPosition*). A Figura 42 apresenta o quadro em RGB, os 1347 pontos dos rosto e essas duas informações sincronizadas.

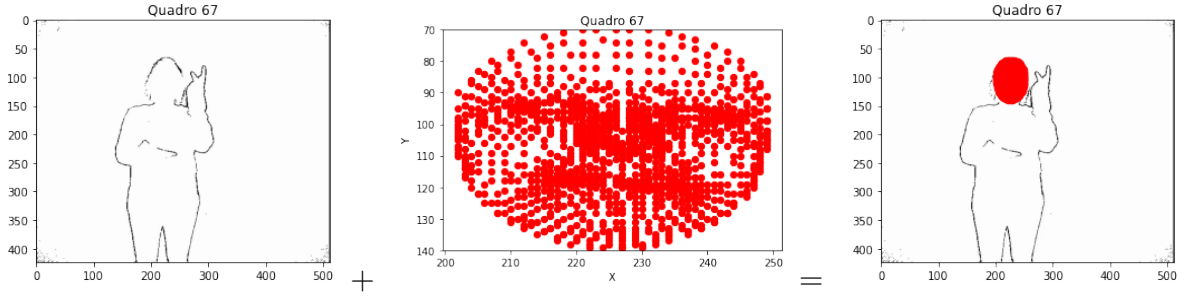
[24] Unidades de animação: 'JawOpen', 'LipPucker', 'JawSlideRight', 'LipStretcherRight', 'LipStretcherLeft', ''LipCornerPullerLeft, 'LipCornerPullerRight', 'LipCornerDepressorLeft', 'LipCornerDepressorRight', 'LeftcheekPuff', 'RightcheekPuff', 'LefteyeClosed', 'RighteyeClosed', 'RighteyebrowLowerer', 'LefteyebrowLowerer', 'LowerlipDepressorLeft', 'LowerlipDepressorRight'.



(a) Quadro RGB (b) 1347 pontos (c) *ColorFaceModel*

Figura 42 – *ColorFaceModel*: posição x-y dos 1347 pontos do corpo plotados no quadro em RGB (Sinalizador 4).

7. *DepthFaceModel*: para esta variável, tem-se os mesmos x-y da *FaceModel*, mas com referência aos quadros em profundidade. Isso significa que a escala se altera, como ilustra a Figura 43.



(a) Quadro em profundidade (b) 1347 pontos (c) *DepthFaceModel*

Figura 43 – *DepthFaceModel*: posição x-y dos 1347 pontos do corpo plotados no quadro em profundidade (Sinalizador 4).

Os dados da face possuem características semelhantes aos do corpo e da mesma forma uma análise deles foi realizada. Quando se diz respeito à informação facial, as abordagens trabalham com o reconhecimento de emoção. Entretanto, a base MINDS-Libras possui sinais cuja a expressão facial é um dos parâmetros fonológicos. Para avaliar a expressão facial, as características mais significativas são: movimentação dos olhos, sobrancelha, testa, queixo, boca e bochecha. Com isso, Microsoft (2014) apresenta 37 pontos[25], destacados na Figura 44, que são classificados com alto nível de detalhamento

[25] Variável e rótulo: EyeLeft = 0, LefteyeInnercorner = 210, LefteyeOutercorner = 469, LefteyeMidtop = 241, LefteyeMidbottom = 1104, RighteyeInnercorner = 843, RighteyeOutercorner = 1117, RighteyeMidtop = 731, RighteyeMidbottom = 1090, LefteyebrowInner = 346, LefteyebrowOuter = 140, LefteyebrowCenter = 222, RighteyebrowInner = 803, RighteyebrowOuter = 758, RighteyebrowCenter = 849, MouthLeftcorner = 91, MouthRightcorner = 687, MouthUpperlipMidtop = 19, MouthUpperlipMidbottom = 1072, MouthLowerlipMidtop = 10, MouthLowerlipMidbottom = 8, NoseTip = 18, NoseBottom = 14, NoseBottomleft = 156, NoseBottomright = 783, NoseTop = 24, NoseTopleft = 151, NoseTopright = 772, ForeheadCenter = 28, LeftcheekCenter = 412, RightcheekCenter = 933, Leftcheekbone = 458, Rightcheekbone = 674, ChinCenter = 4, LowerjawLeftend = 1307 e LowerjawRightend = 1327.

da face. Utilizá-los no lugar dos 1347 pontos disponíveis originalmente possibilita eliminar informações redundantes e reduz o custo computacional dos dados.

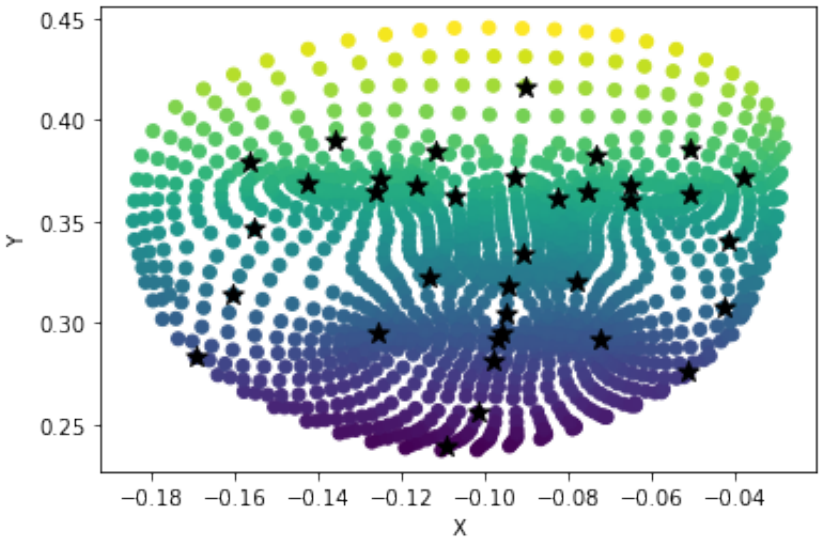


Figura 44 – *FaceModel* e os 37 pontos (preto '*') representativos da face.

3.3.4 Dados do sensor RGB-D: vídeos

O sensor RGB-D disponibilizou vídeos dos sinais em formato ".mp4". Diferentemente dos dados da câmera RGB, nesse caso eles possuem um número fixo de quadros, o mesmo tempo de duração e não houve perda de gravações. Como apresentado na Seção 3.3.1, cada sinalizador tem sua forma própria de executar o sinal, diversificando a velocidade e amplitude do movimento. Manipular essas variáveis é lidar com a realidade presente na comunicação em Libras.

A análise exploratória realizada nos vídeos gravados pelo sensor RGB-D foi relacionada com o início e o fim da execução do sinal. Percebeu-se que havia muitos quadros sem informação de movimento e para trabalhar com esses dados seria interessante tratá-los previamente.

Como o sensor RGB-D gravou diversas informações e todas estão sincronizadas, uma forma para determinar o intervalo de quadros que representam o movimento realizado no sinal foi utilizando a variável *Position* dos pulsos (Pontos 7 e 11). Dessa forma, a velocidade foi calculada para cada mão, respeitando a Equação (3.1), e uma média dos valores foi obtida. Um limiar foi estabelecido zerando o valor da velocidade resultante quando o mesmo era menor do que a média dos valores da amostra em questão. Com isso, foi possível determinar o início e o fim dos vídeos, recortando as partes do vídeo que não possuíam movimento. A Figura 45 ilustra o vetor velocidade para a amostra *2-07Banco_5RGB.mp4* e apresenta que o sinal começa a ser executado no quadro 13 e

termina no quadro 79.

$$Velocidade = \frac{\sqrt{\Delta Position_X^2 + \Delta Position_Y^2 + \Delta Position_Z^2}}{5segundos/150quadros} \tag{3.1}$$

$$\Delta Position = Position_{frame_{i+1}} - Position_{frame_i} \vee i = 1, 2 \ldots 150$$

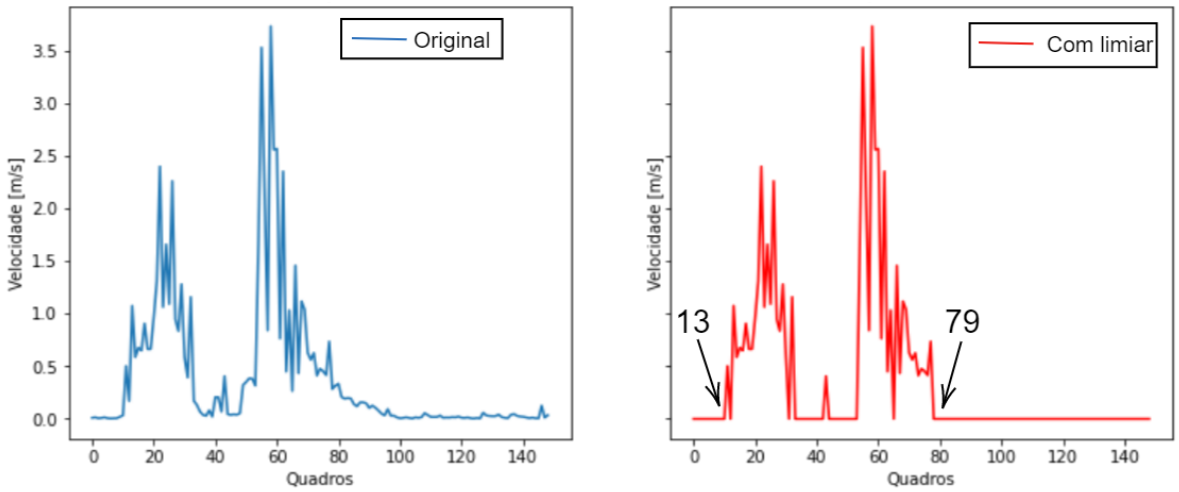


Figura 45 – Velocidade média das mãos na amostra *2-07Banco_ 5RGB.mp4* (Sinalizador: 2, Sinal: Banco, Gravação: 5). Limiar para corte nos quadros 13 e 79.

Essa abordagem é uma das alternativas que permite encontrar o início e o fim de cada vídeo. No caso do sensor RGB-D a implementação é facilitada, pois já se tem o posicionamento das mãos para cada quadro e com isso é possível encontrar a velocidade do movimento. Já para casos em que se tem apenas os vídeos, é necessário aplicar técnicas de distribuição de velocidade em uma cena e suas alterações ao longo do tempo. Um opção pode ser encontrar essa informação utilizando o fluxo óptico (Horn e Schunck, 1981).

3.4 Principais Contribuições do Capítulo

Em suma, este capítulo detalhou todas as características da base MINDS-Libras e realizou uma análise exploratória dos dados. Originalmente a base disponibiliza (i) vídeos em RGB, (ii) vídeos em profundidade, informações de (iii) 25 juntas do corpo e de (iv) 1347 pontos do rosto, que foram capturados por uma câmera RGB e um sensor RGB-D. A base MINDS-Libras é composta por 1200 amostras: 20 sinais gravados 5 vezes por 12 sinalizadores.

O estudo realizado concluiu que:

- Cada sinalizador tem sua forma própria de executar os sinais, mesmo que haja um padrão a ser seguido. A amplitude e a velocidade de execução são os fatores que mais se modificam de pessoa para pessoa, e estão diretamente à prosódia de cada sinal;
- Em relação às juntas do corpo, o movimento realizado pelas mãos tornaram-se mais relevantes e pode ser representado por 10 pontos;
- Em relação ao rosto, 37 pontos podem ser utilizados para caracterizar as expressões faciais devido à sua movimentação significativa;
- Há a necessidade de realizar um pós-processamento nos vídeos para destacar o intervalo de tempo que realmente representa a execução de um sinal.

A base de dados MINDS-Libras e a sua análise descritiva estão disponíveis em Minds (2019) e Rezende (2020), respectivamente.

Capítulo 4

Reconhecimento de Sinais da Libras

Este capítulo apresenta as abordagens utilizadas para o reconhecimento de sinais da Libras. O propósito é que a metodologia elaborada permita o aprendizado dos movimentos realizados na execução de cada sinal, evidenciando os atributos que distinguem cada um deles e evitando a dependência das características físicas do indivíduo que realiza o sinal.

A metodologia geral aplicada neste trabalho seguiu as etapas básicas de um processo de Aprendizado de Máquina, exemplificadas na Figura 46. A coleta de dados é o estágio inicial desse processo e, neste trabalho, é descrita no Capítulo 3 que descreveu a criação da base de dados MINDS-Libras (Minds, 2019). Dado essa primeira etapa, a escolha das técnicas utilizadas em cada um dos próximos estágios dependerão dos dados que serão utilizados para o processo de aprendizagem.

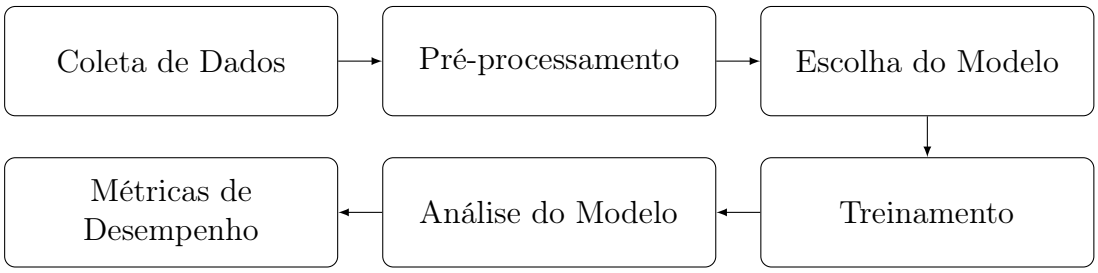


Figura 46 – Etapas básicas de um processo de aprendizado de máquina.

A base de dados desenvolvida neste trabalho, MINDS-Libras, fornece dados de dois sensores: câmera RGB e sensor RGB-D. As informações relativas à câmera foram massivamente utilizadas em Assis (2018), Guerra (2019), Castro et al. (2019), Castro (2020), e Rezende et al. (2021) e, com isso, a ideia principal foi prosseguir nesta pesquisa investigando a contribuição dos dados disponibilizados pelo sensor RGB-D. As abordagens foram estruturadas tendo uma metodologia clássica como base, adequando as técnicas de classificação aos diferentes tipos dados utilizados. Isso significa que foi aplicada uma mesma estratégia para fontes diferentes, com o objetivo de extrair as melhores informações dos dados disponibilizados.

Sendo assim, a Seção 4.1 apresenta a metodologia para reconhecimento dos vídeos em RGB dos sinais da MINDS-Libras. Essa abordagem tem como referência Rezende et al. (2021), na qual foi realizada algumas mudanças na metodologia para melhorar o desempenho do modelo e o seu poder de generalização. Em seguida, a Seção 4.2 descreve uma abordagem que considera a trajetória das mãos para o reconhecimento do sinal. Nesse experimento foi implementada uma metodologia independente do sinalizador, focando no movimento realizado durante a execução do sinal. A base para a concepção dessa abordagem foi a análise exploratória realizada nos dados (Seção 3.3) e que possibilitou a visualização de características significativas para o reconhecimento dos sinais.

4.1 Abordagem utilizando vídeos gravados em padrão RGB

Com a base de dados descrita no Capítulo 3, uma metodologia para classificação dos sinais foi definida. Os experimentos realizados em Castro et al. (2019) e Rezende et al. (2021) foram o ponto de partida para a abordagem apresentada nesta seção, sendo implementadas algumas modificações, como mostra o Quadro 1:

Quadro 1 – Alterações realizadas na abordagem de Rezende et al. (2021)

Em Rezende et al. (2021)	**Modificações realizadas**
Sumarização aplicada no vídeo da base de dados	Sumarização aplicada no vídeo após análise dos quadros com movimento
10 quadros significativos retornados da sumarização	5 quadros significativos retornados da sumarização
Divisão aleatória $75\% - 25\%$ para treino e teste, por sinal	Validação cruzada, com 12-folds, por sinal

Essas alterações permitiram que a abordagem aqui apresentada (i) recebesse dados com informações significativas sobre o sinal, uma vez que os quadros iniciais e finais dos vídeos sem movimento foram descartados; (ii) processasse dados representativos, sendo cinco quadros suficientes para caracterizar o sinal; e (iii) garantisse que as amostras participassem ora do conjunto de treino, ora do de teste, por meio da validação cruzada.

4.1.1 Pré-processamento

O primeiro pré-processamento realizado foi o corte inicial e final dos vídeos, conforme apresentou a Seção 3.3.4. Os quadros foram eliminados por meio da análise de velocidade das mãos que demonstrou o intervalo de tempo que o sinal está sendo executado. Em uma das amostras, 83 dos 150 quadros foram eliminados (Quadros 1 a 12 e 80 a 150), sendo possível perceber pelas Figuras 47 e 48 que não há movimento executado nas imagens.

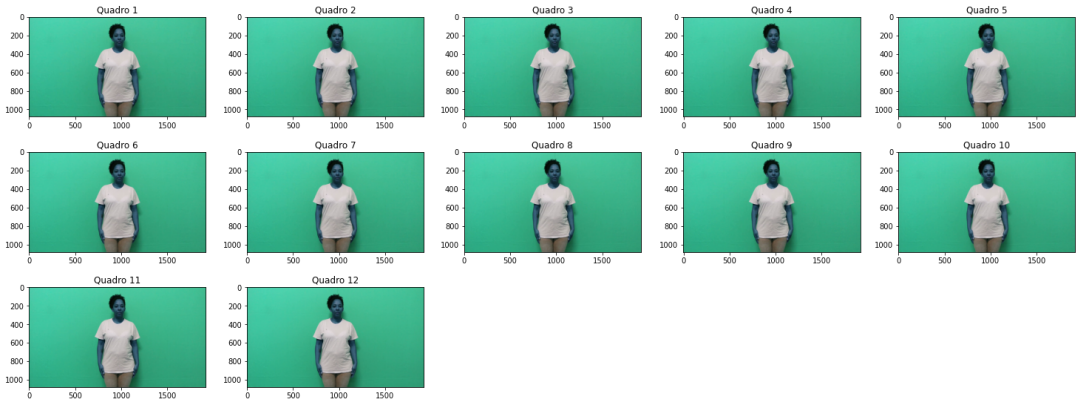


Figura 47 – Quadros iniciais (1 a 12) da amostra *6-01Acontecer_2RGB.mp4* (Sinalizador: 6, Sinal: Acontecer, Gravação: 2) que não caracterizam movimento.

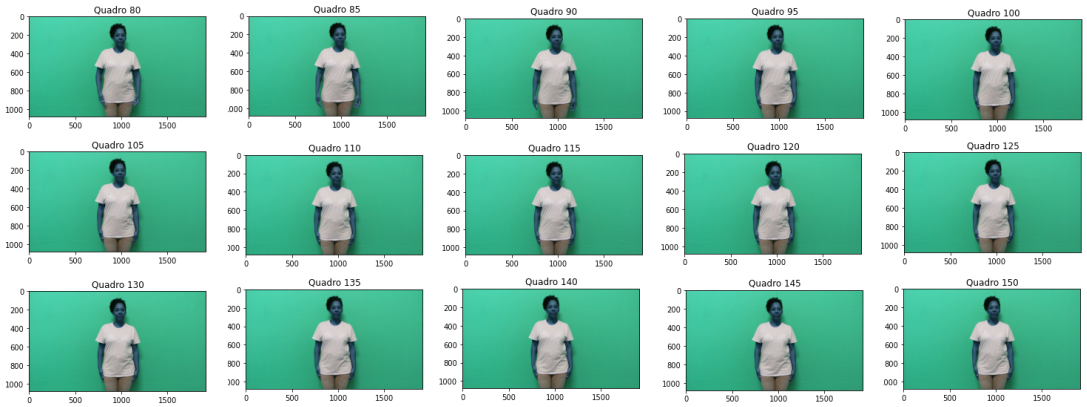


Figura 48 – Quadros finais (80 a 150, de 5 em 5) da amostra *6-01Acontecer_2RGB.mp4* (Sinalizador: 6, Sinal: Acontecer, Gravação: 2) que não caracterizam movimento.

Mesmo eliminando o início e o fim de cada amostra, os vídeos resultantes desse processamento apresentaram quadros com informações similares às dos seus vizinhos. Isso faz com que não haja elemento que os torne distintos e que seja significante para diferenciar aquele vídeo dos demais. Essa característica das amostras se deve a velocidade de execução do sinal que se difere de sinalizador para sinalizador e faz com que na maioria dos casos a taxa de captura do sensor registre uma sequência de quadros com mesma informação. Como manter essas imagens resultaria no processamento de dados redundantes, optou-se por realizar a sumarização, tornando o processamento mais eficiente e menos custoso computacionalmente. De acordo Almeida (2014b) com esse pré-processamento reduz-se o tamanho dos vídeos com o mínimo de perda de informação possível.

Dentre as várias técnicas de sumarização encontradas na literatura, a abordagem implementada em Freitas et al. (2014a) e Almeida et al. (2015) foi a utilizada nessa abordagem. Nesses trabalhos, os autores abordaram o Problema da Diversidade Máxima (PDM) (Kuo et al., 1993) que, quando aplicado para sumarização de vídeos, baseia-se na diversidade entre o quadro m e o quadro n, isto é, na máxima distância temporal e

na máxima diferença de cores RGB entre eles. Para resolver o PDM foi empregada a estratégia evolutiva denominada MSES (*Memetic Self-Adaptive Evolution Strategies*).

Como a sumarização reduz o número de quadros de cada sinal, fez-se necessário escolher qual será o número final de imagens que irá representar cada amostra. Essa operação padroniza o tamanho das amostras e garante que os vetores de características tenham o mesmo tamanho. Entretanto, não existe um número padrão que pode ser aplicado a esse tipo de caso. Almeida (2014b); Almeida et al. (2014); Rezende (2016); Rezende et al. (2017); Guerra et al. (2018) selecionaram 5 quadros para representar cada sinal, em Pan et al. (2020b) foram 8, Castro et al. (2019) e Rezende et al. (2021) utilizaram 10, já Al-Hammadi et al. (2020a) optaram por 16. Esses valores foram suficientes para obter taxas de reconhecimento satisfatórias tendo em vista o que foi proposto em cada estudo. Contudo, verificou-se por meio de análise visual e experimental[1] que dentre o número de quadros disponíveis, 5 imagens seriam suficientes sem acarretar na perda de informações de movimento, como ilustram as Figuras 49 e 50. Analisando a sequência de imagens apresentadas na Figura 50 é notório que há muitos quadros semelhantes fazendo com que a sumarização seja uma etapa necessária para o processamento dos vídeos.

Figura 49 – Quadros 22, 38, 48, 67 e 79 retornados da sumarização aplicada na amostra *2-07Banco_5RGB.mp4* (Sinalizador: 2, Sinal: Banco, Gravação: 5).

Prosseguindo na etapa de pré-processamento, as imagens retornadas da sumarização foram cortadas e redimensionadas. O corte foi realizado nas imagens alterando a configuração inicial de 1920×1080 para 1080×1080 *pixels*, com o intuito de eliminar parte do *background* nas laterais, reduzindo a quantidade de informação a ser processada, e de adequar os quadros para a CNN, que recebe imagens quadradas. Não foi necessário nessa etapa enquadrar a posição do sinalizador, pois o mesmo já tinha uma posição centralizada ao gravar os vídeos. Em seguida elas foram convertidas para a escala de cinza e redimensionadas na proporção 224×224 *pixels*, valor esse escolhido tomando como base as dimensões de entrada de uma rede VGG16[2] (Simonyan e Zisserman, 2014b). Inicialmente o intuito era utilizar essa rede para classificar os sinais da Libras, mas testes realizados concluíram que uma rede com menos camadas seria suficiente devido a quantidade de sinais e permitiria um processamento menos custoso.

[1] O número de quadros utilizado nesta tese foi obtido a partir de testes empíricos, entretanto, vale ressaltar que, se o número de classes aumentar, provavelmente o número de quadros significativos deve se alterar para que eles sejam representativos. Além disso, a rede de classificação também precisará ser revisada.

[2] A VGG16 possui 16 camadas convolucionais, camada de *MaxPooling* e *Softmax*. A imagem de entrada dessa rede possui dimensão de 224×224 *pixels*.

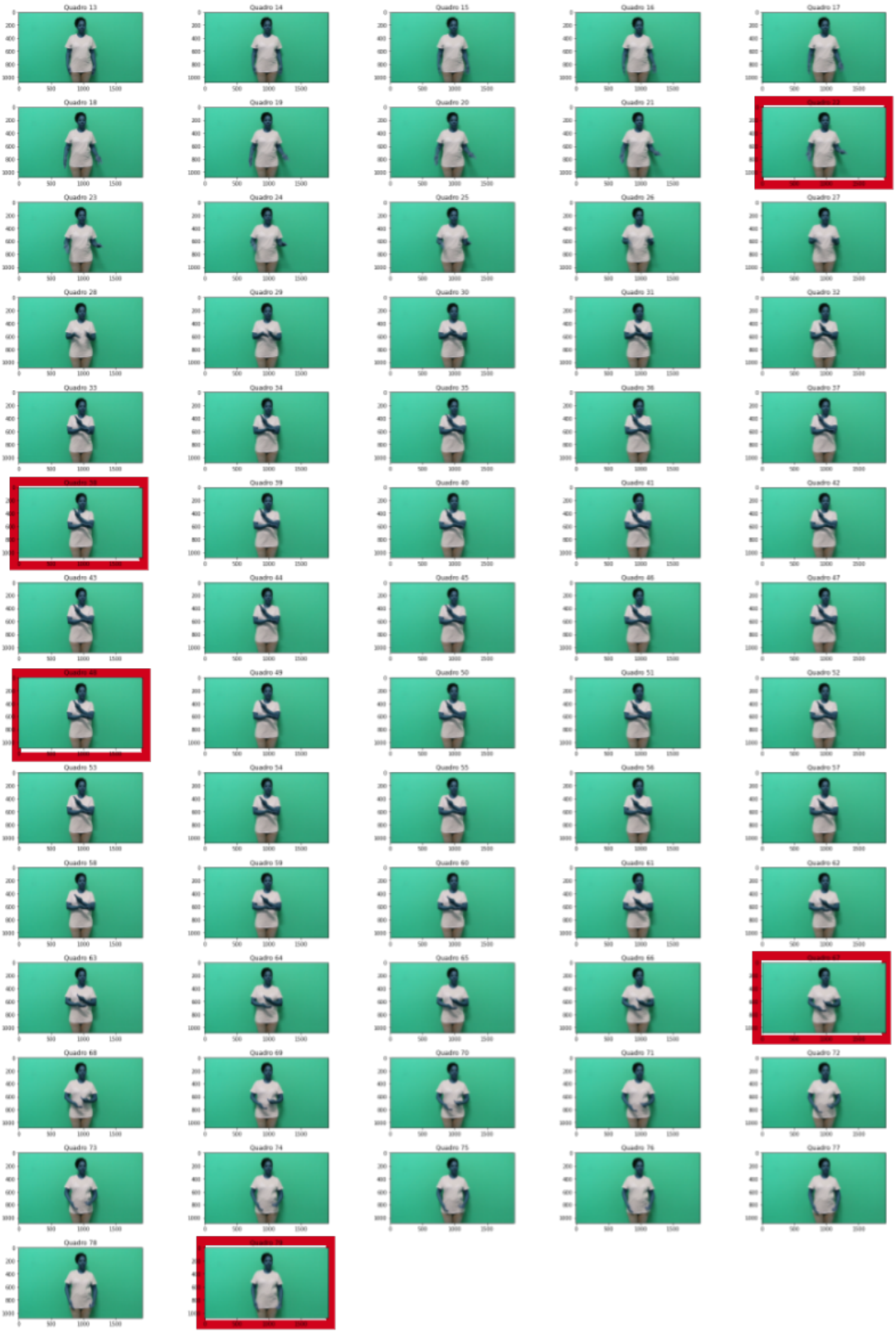


Figura 50 – Quadros com movimento (13 ao 79) da amostra *2-07Banco_ 5RGB.mp4* (Sinalizador: 2, Sinal: Banco, Gravação: 5) e os 5 quadros retornados da sumarização, em destaque: 22, 38, 48, 67 e 79.

A última operação realizada para tratar os dados foi o *data augmentation*, cuja premissa é ter um conjunto de dados retirado de um conjunto limitado de condições (Gandhi, 2018). Há várias técnicas para aumentar dados com alterações espaciais e/ou temporais: a primeira inclui operações de translação, espelhamento horizontal e redimensionamento das imagens (Krizhevsky et al., 2012b; Simonyan e Zisserman, 2014a) e a segunda envolve translação temporal, escalonamento da duração da sequência e deformação no domínio do tempo (Pigou et al., 2014; Molchanov et al., 2015). O *data augmentation* pode ser aplicado neste trabalho pois a CNN classifica dados de maneira robusta, mesmo que eles sejam colocados em orientações diferentes, devido a sua propriedade de invariância (Gandhi, 2018), além de aumentar os dados a partir do conjunto inicial que se tem acesso, tendo em vista que a criação e a expansão de uma base de dados de sinais da Libras não é uma tarefa trivial. As seguintes técnicas temporais e espaciais, para aumentar os dados, aplicadas a cada um dos 5 quadros nessa abordagem foram:

1. Espelhamento horizontal das imagens; e
2. Zoom aleatório de 5% a 15% em cada quadro.

4.1.2 Escolha do modelo: arquitetura CNN3D

De posse dos dados pré-processados, a próxima ação foi desenvolver uma rede capaz de realizar a extração de características e a sua classificação. Como não há uma configuração padrão que resolva este problema e dentre as diversas arquiteturas presentes na literatura, esta seção expõe as escolhas realizadas, após vários experimentos e decisões tomadas.

Nesta abordagem, o objetivo foi reconhecer sinais da Libras por meio de uma sequência de imagens com lógica temporal. De acordo com Aghdam e Heravi (2017), a CNN é a técnica mais indicada quando se deseja resolver esse tipo de problema. Como detalhado em Rezende (2019), as Redes Neurais Convolucionais fazem parte da área de Aprendizado Profundo e podem ser definida como uma Rede Neural com expansão da quantidade de camadas intermediárias, fazendo com que cada camada seja responsável por aprender características que auxiliem no processo de classificação.

Em suma, a arquitetura da CNN envolve três tipos de camadas: convolucional, *pooling* (redução) e totalmente conectada (*Fully Connect - FC*). Elas podem ser ajustadas, combinadas e intercaladas, permitindo a criação de uma multiplicidade de *frameworks*. Essas variações de CNN's tornaram a técnica competitiva no âmbito da Visão Computacional, tendo em vista que há arquiteturas de CNN já consolidadas para resolver problemas específicos. Entretanto, para que a CNN seja capaz de obter a informação temporal dos quadros que compõem os vídeos dos sinais da Libras, utilizaram-se convoluções 3D nas camadas convolucionais (Ji et al., 2012; Tran et al., 2014). Estudos realizados por Tran

et al. (2014), indicam a habilidade das CNN's 3D em captar tais características. Dessa forma, neste trabalho ela foi utilizada e é caracterizada por 4 camadas de convolução, sendo cada uma delas seguida por uma função de ativação ReLU[3] e por uma camada de *MaxPooling*[4], como apresentado na Figura 51 e detalhado na Tabela 8.

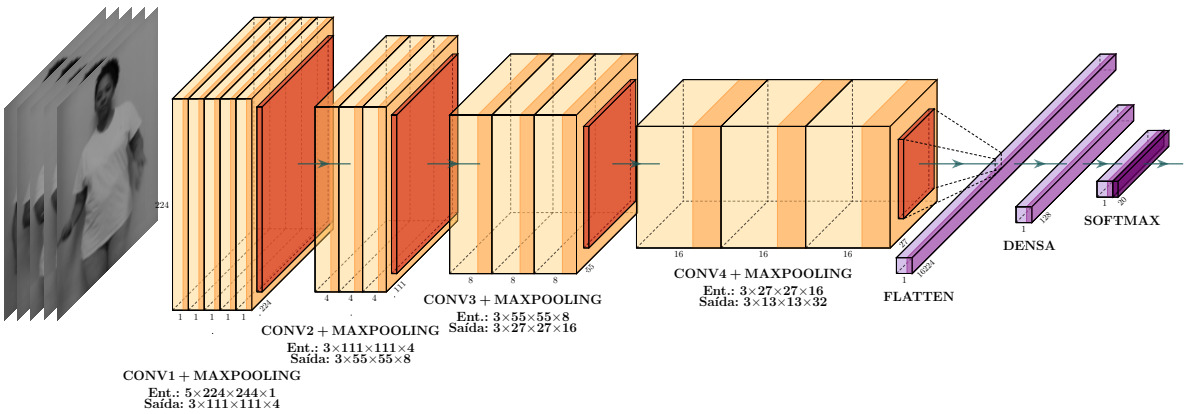


Figura 51 – Arquitetura da CNN 3D proposta.

Tabela 8 – Parâmetros da arquitetura.

Descrição	Entrada	Kernel	Filtros	Saída
Volume de entrada	-	-	-	$5 \times 224 \times 224$
Conv3D_1	$5 \times 224 \times 224$	$3 \times 3 \times 3$	4	$3 \times 222 \times 222 \times 4$
MaxPool3D	$3 \times 222 \times 222 \times 4$	$1 \times 2 \times 2$	-	$3 \times 111 \times 111 \times 4$
Conv3D_2	$3 \times 111 \times 111 \times 4$	$3 \times 3 \times 3$	8	$3 \times 111 \times 111 \times 8$
MaxPool3D	$3 \times 111 \times 111 \times 8$	$1 \times 2 \times 2$	-	$3 \times 55 \times 55 \times 8$
Conv3D_3	$3 \times 55 \times 55 \times 8$	$3 \times 3 \times 3$	16	$3 \times 55 \times 55 \times 16$
MaxPool3D	$3 \times 55 \times 55 \times 16$	$1 \times 2 \times 2$	-	$3 \times 27 \times 27 \times 16$
Conv3D_4	$3 \times 27 \times 27 \times 16$	$3 \times 3 \times 3$	32	$3 \times 27 \times 27 \times 32$
MaxPool3D	$3 \times 27 \times 27 \times 32$	$1 \times 2 \times 2$	-	$3 \times 13 \times 13 \times 32$
Flatten	$3 \times 13 \times 13 \times 32$	-	-	5408
Fully Connected	5408	-	-	128
Dropout	128	-	-	128
Fully Connected	128	-	-	20

A entrada da CNN é um volume de dimensão $5 \times 224 \times 224$, indicando os 5 quadros retornados do pré-processamento, cujas proporções são 224×224 em um canal (Escala de cinza). Esse volume de entrada passa, quatro vezes, por um camada convolucional (Conv3D), seguida por função de ativação ReLU e uma camada de *MaxPooling* (MaxPool3D). O filtro utilizado nessas camadas foi o $3 \times 3 \times 3$[5], sendo essa configuração escolhida devido ao bom desempenho mostrado por Tran et al. (2014). As camadas convolucionais possuem 4, 8, 16, e 32 filtros, respectivamente, responsáveis pela profundidade dos mapas de características.

[3] Função de ativação ReLU: $f(z) = max(z, 0)$
[4] *MaxPooling*: operação de convolução que calcula o valor máximo entre os *pixels*.
[5] As dimensões do filtro representam Profundidade×Largura×Altura.

As camadas de *MaxPooling*, que também realizam operações em profundidade, possuem *kernels* $1 \times 2 \times 2$.

A Tabela 8 apresentou a arquitetura da rede. Do volume de entrada para a primeira camada convolucional houve uma alteração nas dimensões resultando em uma saída $3 \times 222 \times 222 \times 4$. Os três primeiros valores foram reduzidos em duas unidades pois não houve *padding*[6] e a última dimensão foi inserida, pois ela faz referência ao filtro aplicado. Na segunda, terceira e quarta camadas convolucionais, aplicou-se *padding* com zeros e passo[7] de dimensão $1 \times 1 \times 1$ para que os mapas de características da saída da convolução tenham as mesmas dimensões dos mapas de entrada, sendo reduzidos pela metade somente pelas camadas de *pooling*[8], quando o filtro $1 \times 2 \times 2$ foi aplicado.

Após a parte convolucional, aplica-se o *flatten*[9] (achatamento) para transformar o volume em um vetor e passa-se à parte totalmente conectada da rede. Foram utilizadas duas camadas totalmente conectadas, sendo a primeira constituída por 128 neurônios com a função de ativação ReLU. Nela foi aplicada a técnica de regularização *dropout*, que remove temporariamente e aleatoriamente alguns neurônios dessa camada durante o treinamento da rede, forçando neurônios a aprenderem de forma mais robusta e independente da configuração de outros neurônios da rede (Hinton et al., 2012). De acordo com Dertat (2017) mesmo em modelos ditos estado da arte, o *dropout* possibilita um ganho substancial, sendo usado para prevenir *overfitting*[10]. Já a segunda camada totalmente conectada possui 20 neurônios e a função *softmax*[11] funcionando como um classificador. A saída da CNN 3D é a saída dessa função de ativação, que é um vetor de 20 posições indicando a probabilidade de a entrada pertencer a cada uma das 20 classes de sinais.

4.1.3 Treinamento

As 1200 amostras originais foram divididas em $12-$*folds*, 11 para o processo de treinamento e 1 para teste (= 100 amostras). O conjunto de treinamento passou pelo *data augmentation* (1100 amostras originais + 2200 geradas sinteticamente) e 3200 das suas amostras foram utilizadas para treino e 100 para validação. Essa divisão foi feita por sinal, indicando que os sinalizadores estão presentes nas amostras de treino e de teste, e os sinais estão balanceados em ambos conjuntos. Os valores de cada conjunto que, inicialmente variavam de 0 a 255, foram padronizados de 0 a 1.

Para o treinamento da rede foi definida uma taxa de aprendizado inicial de $0,001$, ajustada pelo otimizador *Adam*, e a inicialização dos pesos foi feita de forma

[6] *Padding*: preenchimento das bordas da imagem com o valor zero ou repetindo os valores da borda.
[7] Passo: descreve quantos *pixels* cada deslocamento percorre.
[8] *Pooling*: camada tem a função de reduzir a dimensão dos mapas de características, reduzindo, assim, a quantidade de parâmetros da rede.
[9] *Flatten*: esta camada é a divisão entre a parte de extração de características e a classificação.
[10] *Overfitting*: sobreajuste aos dados.
[11] *Softmax*: camada de classificação que calcula a probabilidade das amostras pertencerem a cada classe.

aleatória. A função de perda foi a *categorical_ crossentropy* e a métrica para avalizar o conjunto de validação foi a acurácia, escolhas comuns em problemas de classificação que indica o desempenho do modelo. Foram utilizados lotes de tamanho 128 e o treinamento interrompido após 50 épocas. Como ilustra a Figura 52, o experimento foi realizado 12-folds, sendo obtidas métricas de desempenho médias. A implementação foi feita em Python, usando o ambiente Google Colab, com os *frameworks* NumPy, Pandas, cv2, PyPlot, Sklearn, Keras e TensorFlow.

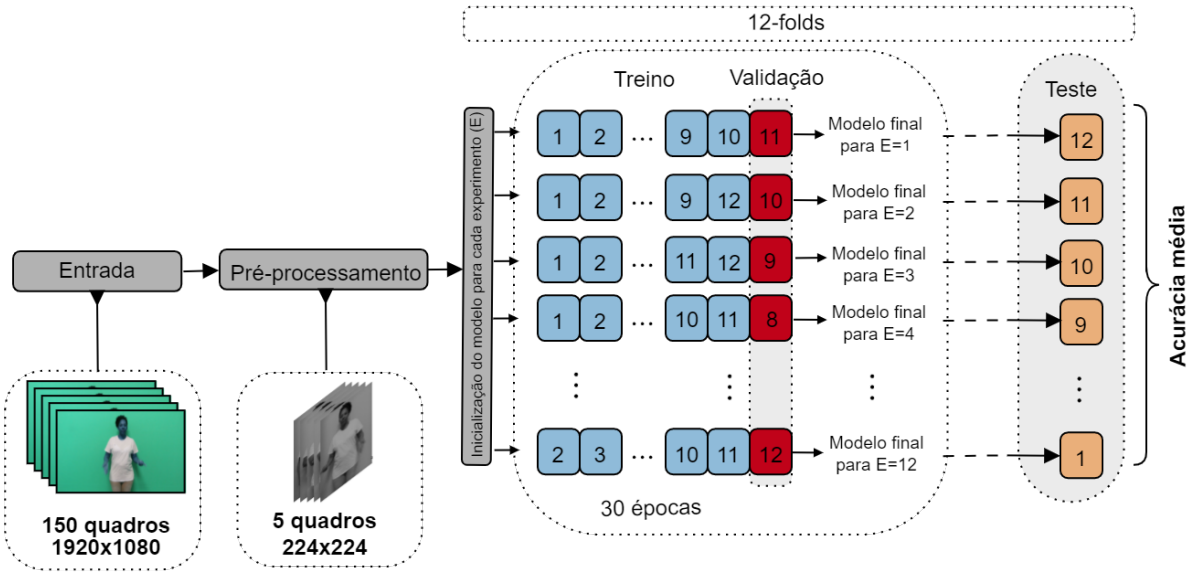


Figura 52 – Treinamento do modelo CNN3D.

4.1.4 Análise do modelo

As CNN's podem ser aplicadas na classificação e predição de diversos problemas, pois permitem variações e combinações de seus parâmetros. Encontrar uma arquitetura depende de testes exaustivos e da análise de trabalhos presentes na literatura para embasar as escolhas realizadas.

A estrutura aqui apresentada foi definida com base em Castro et al. (2019) e Rezende et al. (2021), sendo realizados alguns experimentos para verificar se haveria necessidade da alteração de algum parâmetro da rede. As principais variações foram: (i) na quantidade de quadros entre as camadas convolucionais, utilizando *kernels* de profundidade diferentes (*Kernel* = Profundidade×Largura×Altura); (ii) troca da operação de *MaxPooling* para *AvaregePooling*[12]; (iii) alteração na quantidade de neurônios na FC e (iv) na quantidade de filtros das camadas convolucionais; (v) retirada do *dropout* e (vi) da primeira camada de FC; (vii) variação na quantidade de quadros retornados da sumarização e (viii) inclusão de um canal com a informação de profundidade. Esses testes

[12] *AvaregePooling*: operação de convolução que calcula o valor médio entre os *pixels*.

permitiram a definição da topologia da rede já apresentada e resultaram nas modificações apresentadas no Quadro 1, cujo critério foi a taxa de acerto da classificação.

4.2 Abordagem utilizando a informação das mãos

A abordagem utilizando a trajetória das mãos foi estruturada após a realização da análise exploratória da base de dados MINDS-Libras, descrita na Seção 3.3. O intuito foi desenvolver uma metodologia independente do sinalizador que permitisse o aprendizado do movimento realizado ao executar o sinal, sem que o modelo ficasse dependente das características físicas do sinalizador. A variação dos pontos ao longo do tempo foi comparada com o comportamento de séries temporais e, dessa forma, foi escolhida a arquitetura TCN para a classificação dos dados. Essa abordagem é desafiadora, pois mesmo com um protocolo de gravação padronizado a execução do sinal possui variação de velocidade e amplitude de um sinalizador para o outro. Essa é uma característica inerente de cada indivíduo e a metodologia foi estruturada para explorar alternativas que contornem essas disparidades.

4.2.1 Pré-processamento

Os dados de entrada do modelo de classificação foram estruturados a partir dos cinco pontos de ambas as mãos para representar o movimento do sinal: cotovelos (Pontos 6 e 10), pulsos (Pontos 7 e 11), mãos (Pontos 8 e 12), pontas dos dedos (Pontos 22 e 24) e polegares (Pontos 23 e 25). Como não houve calibração dos sensores na criação da MINDS-Libras foi necessário realizar um pré-processamento nos dados brutos para padronizar as amostras. Para cada quadro, o valor da coordenada de cada um dos 10 pontos foi subtraído da coordenada que representa a cabeça (Ponto 4 - Cabeça). Isso é ilustrado pelas Equações (4.1), (4.2) e (4.3), onde $x_{rp,q}$, $y_{rp,q}$ e $z_{rp,q}$ são as coordenadas x-y-z relativas do ponto p no quadro q; $x_{p,q}$, $y_{p,q}$ e $z_{p,q}$ são as coordenadas x-y-z de p em q (dados brutos); $x_{ponto4,q}$, $y_{ponto4,q}$ e $z_{ponto4,q}$ são as coordenadas x-y-z de ponto da cabeça em q.

$$\underbrace{x_{rp,q}}_{\text{coordenada relativa de p, quadro q}} = \underbrace{x_{p,q} - x_{ponto4,q}}_{\vee\ \text{p} = 6,7,8,10,11,12,22,23,24,25} \tag{4.1}$$

$$\underbrace{y_{rp,q}}_{\text{coordenada relativa de p, quadro q}} = \underbrace{y_{p,q} - y_{ponto4,q}}_{\vee\ \text{p} = 6,7,8,10,11,12,22,23,24,25} \tag{4.2}$$

$$\underbrace{z_{rp,q}}_{\text{coordenada relativa de p, quadro q}} = \underbrace{z_{p,q} - z_{ponto4,q}}_{\vee\ \text{p} = 6,7,8,10,11,12,22,23,24,25} \tag{4.3}$$

Em seguida, as coordenadas foram normalizados entre $[0, 1]$, por eixo e por quadro. No final, os valores foram unidos, como mostrado na Figura 53, resultando em uma matriz de entrada com 30 séries temporais de 150 valores.

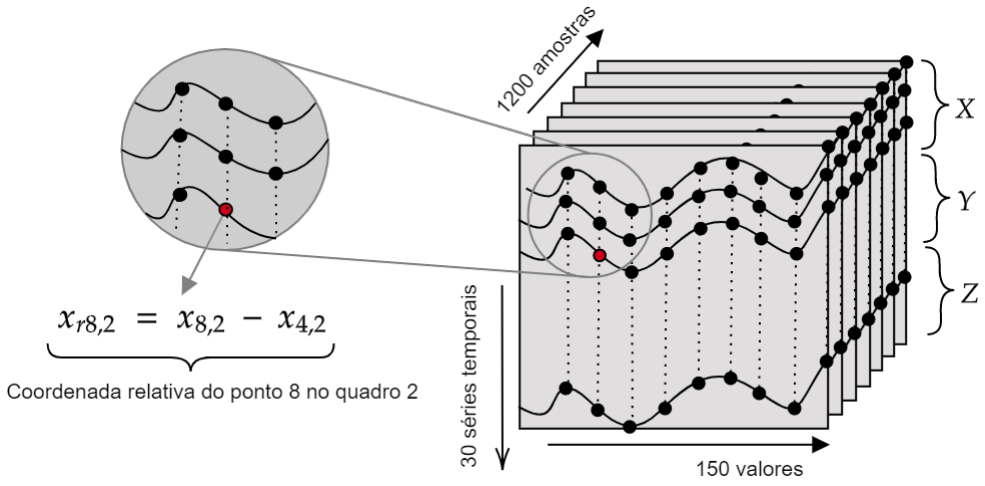


Figura 53 – Matriz de entrada com 30 séries temporais (coordenadas x-y-z) em 150 quadros para cada uma das 1200 amostras. Em destaque: equação referente à terceira série temporal (coordenada x, $p = 8$) do segundo quadro ($q = 2$) na primeira amostra.

Para que o modelo receba uma variação maior de amostras em termos de velocidade e amplitude, esta abordagem aplicou técnicas de *data augmentation*. Elas servem para reduzir o *overfitting* e expandir o limite de decisão dos modelos de Aprendizado de Máquina usando padrões sintéticos. Com isso, amostras artificiais foram geradas tentando aumentar a representatividade das variações ilustradas graficamente na Figura 38 (Seção 3.3.2). Como os dados representam séries temporais, as técnicas empregadas foram as já utilizadas para esse tipo de dado, com base em Le Guennec et al. (2016):

1. Para a variação de velocidade de execução do sinal, a técnica de deformação da janela (*Warp*) foi utilizada, expandindo ou contraindo aleatoriamente as janelas dos sinais;

2. Para gerar curvas que começam em momentos diferentes, a técnica *Shift* foi usada, atrasando ou acelerando o sinal, aplicando uma variável aleatória com distribuição gaussiana (média 0 e desvio padrão de 5 quadros).

3. Para minimizar o ruído das amostras, a suavização de curva também foi usada com tamanho de janela igual a 7 (Suavização_W7).

A Figura 54 ilustra cada uma das técnicas descritas. Inicialmente o *data augmentation* foi aplicado a todas as amostras da base, mas o algoritmo só usou os relativos aos dados de treinamento.

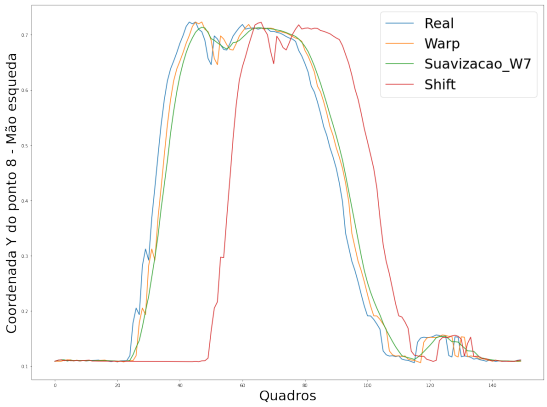


Figura 54 – *Data augmentation* relativo à série temporal do ponto 8-MãoEsquerda (coordenada Y) da amostra *1-01Acontecer_ 1Body.txt*.

4.2.2 Escolha do modelo: arquitetura TCN

De acordo com Asadi-Aghbolaghi et al. (2017), a adição da informação temporal no reconhecimento de ações e gestos tornou o problema desafiador devido ao processamento dos dados e à complexidade dos modelos. Nesse contexto, estratégias como a sub-amostragem de quadro, adotada na abordagem anterior, e modelagem da sequência temporal com as Redes Neurais Recorrentes (RNN), por exemplo, são alternativas adotadas para contornar esta questão. Entretanto, Bai et al. (2018) mostraram que a Rede Convolucional Temporal (TCN) é capaz de superar implementações padrão de RNN. No artigo, os autores verificaram que uma arquitetura convolucional simples como a TCN supera redes recorrentes como as LSTM's (*Long-Short-Term Memory*) em diversas tarefas e conjuntos de dados, ao mesmo tempo em que demonstra memória mais eficaz.

Outro ponto que direcionou a escolha da TCN como classificador da trajetória manual foi a natureza dados. A primeira referência dessa rede foi o trabalho de Lucas et al. (2020) que apresentaram a TCN para previsão de séries temporais de evapotranspiração. Apesar dessa técnica não ter sido vista em trabalhos de reconhecimento automático de sinais da Libras, ela teve um bom desempenho para problema de previsão citado e em Lin et al. (2019) que realizaram a classificação de dados médicos. Dessa forma, pela aplicação ter uma variável que tem alteração ao longo do tempo, buscou-se investigar o desempenho dessa técnica.

Em suma, os princípios do TCN são: (i) sua arquitetura é totalmente convolucional, isto é, a rede produz uma saída do mesmo comprimento que a entrada e (ii) utiliza convoluções causais, o que significa que não há vazamento de informações do futuro para o passado. Como detalhado em Lucas et al. (2020), a rede é composta por blocos residuais que são formados por camadas convolucionais causais dilatadas[13], seguidas da função de

[13] Uma convolução causal dilatada significa que o modelo não viola a ordenação na qual os dados foram modelados e que na convolução o filtro é aplicado sobre uma área maior que seu comprimento, pulando valores de entrada com uma certa etapa (Lucas et al., 2020).

ativação ReLU, normalização dos pesos dos filtros e *dropout*.

A arquitetura utilizada nesta abordagem é ilustrada pela Figura 55. A entrada da TCN são matrizes que contém as 30 séries temporais de 150 valores, exemplificadas pela Figura 53. Esses dados passaram, em sequência por três blocos residuais, um com fator de dilatação $d = 1$ $(= 2^0)$, 2 $(= 2^1)$ e 4 $(= 2^2)$. Esses blocos, por sua vez, são formadas por duas camadas convolucionais causais dilatadas, seguidas da normalização dos pesos dos filtros (*WeightNorm*) e a função de ativação ReLU. Para garantir que as larguras de entrada e saída não sejam discrepantes, uma convolução adicional de tamanho 1×1 foi usada nesse bloco. Ao final desse processo, há uma camada densa com a saída definida por meio da função *Softmax*.

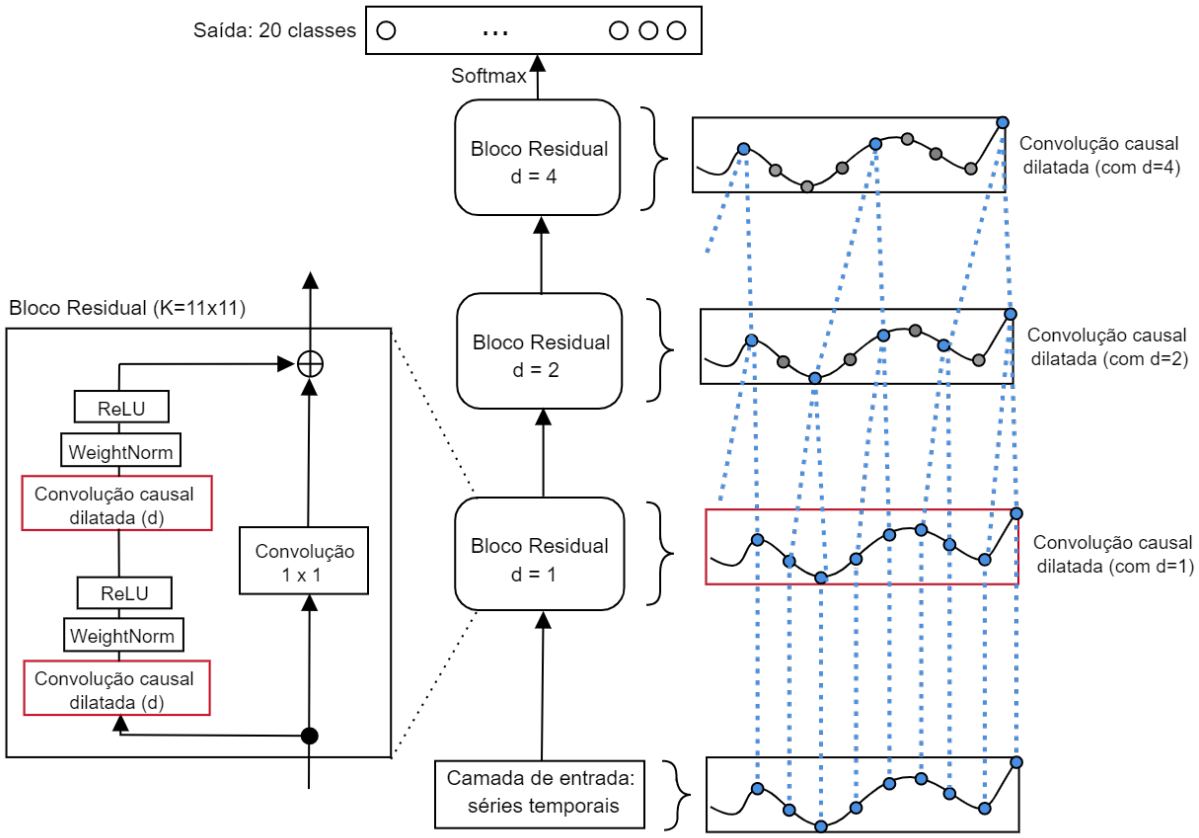


Figura 55 – Arquitetura TCN utilizada.

4.2.3 Treinamento

Em um contexto real na execução de sinais da Libras, cada pessoa tem a própria maneira de realizar o sinal, sendo essa uma característica intrínseca do indivíduo. Essas variações podem ser expressas pela mudança de velocidade, movimento e amplitude da execução. Dessa forma, esta abordagem buscou um modelo capaz de aprender o movimento realizado pelo sinalizador em vez das propriedades físicas do mesmo. Para isso, o conjunto de treino e teste foram divididos pelo sinalizador.

Dos 12 sinalizadores presentes na base de dados, 11 foram utilizados para treinamento (= 1100 amostras) e 1 para teste (= 100 amostras). O conjunto de treino passou pela etapa de *data augmentation*, aumentando o volume de dados em três vezes, o que resultou em 3300 amostras. Os pesos foram inicializados com zeros, o otimizador foi o *Adam* e o treinamento foi interrompido após 100 épocas. O experimento foi realizado 30 vezes por sinalizador, sendo obtidas métricas de desempenho médias, como ilustra a Figura 56. A implementação foi feita em Python, usando o ambiente Google Colab, com os *frameworks* NumPy, Pandas, Sklearn, Keras e TensorFlow.

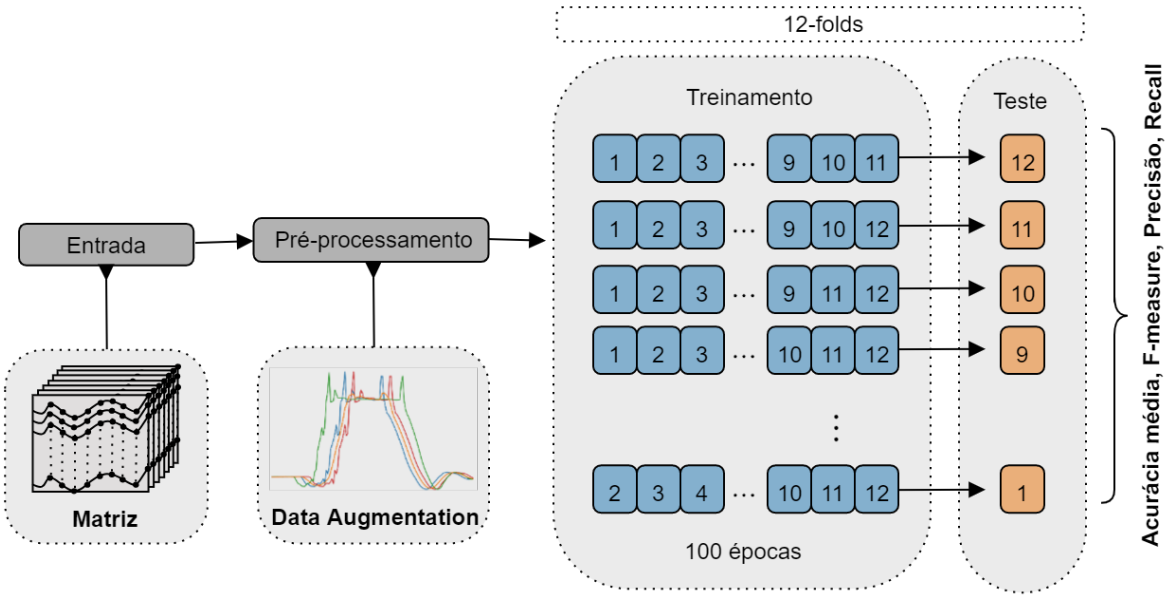


Figura 56 – Treinamento do modelo TCN.

Para finalizar os experimentos desta abordagem, a divisão dos conjuntos de treino e teste foi realizada por sinal, sendo que a proporção de amostras por *fold* foi mantida. Esse experimento teve como objetivo confrontar os resultados com relação ao *baseline* apresentado na Seção 4.1.

4.2.4 Análise do modelo

A arquitetura do modelo utilizada foi apresentada na Figura 55. Para obter aquela estrutura foi realizada uma otimização dos hiperparâmetros da rede com um algoritmo genético Vanilla (Bai et al., 2018), como em Lucas et al. (2020). O espaço de busca e os valores finais adotados na implementação foram descritos na Tabela 9. Além disso, experimentos com o uso de *ensemble learnings* combinados pela moda das saídas de cinco modelos de TCN's foram analisados, mas os resultados não apresentaram melhora no desempenho.

Tabela 9 – Determinação dos parâmetros da TCN.

Parâmetros	**Valores adotados**	**Espaço de busca**
Número de filtros na camada de convolução	16	[16,32,64]
Tamanho do kernel	11 x 11	[2,3,5,11]
Porcentagem de *dropout*	0	[0 ... 0.5]
Normalização do batch	Sim	Sim, Não
Número de dilatações causais	2	[1 ... 5]
Número de pilhas de blocos	1	[1,2]

4.3 Principais Contribuições do Capítulo

Este capítulo detalhou os experimentos realizados para classificar os sinais da base de dados MINDS-Libras. A primeira abordagem é resultante dos trabalhos de Castro et al. (2019) e Rezende et al. (2021), com propostas de melhorias nas metodologias que já tinham sido aplicadas previamente. O objetivo foi investigar o impacto na mudança de alguns parâmetros e apresentar uma metodologia para o reconhecimento de sinais com a CNN 3D que pode ser utilizada em qualquer outra base de dados que possua vídeos de sinais da Libras ou de outra língua. Já o foco da segunda abordagem foi no reconhecimento do sinal utilizando as informações manuais. Como os dados representam uma série temporal, resolveu-se investigar o desempenho da TCN para classificá-los com o foco em uma metodologia independente do sinalizador. A Tabela 10 apresenta um resumo dos experimentos e a implementação deles está disponível em Rezende (2020).

Tabela 10 – Características dos experimentos com a base de dados MINDS-Libras

Experimento	Entrada	Pré-processamento	Abordagem	Modelo	Épocas
1	Escala de Cinza	S/R: 5 quadros	Por sinal	CNN3D	30
	Escala de Cinza	$\hookrightarrow$ + DA	Por sinal	CNN3D	50
2	*Position* x-y-z	DA	Por sinal	TCN	100
3	*Position* x-y-z	DA	Por sinalizador	TCN	100

*S/R: Sumarização e Redimensionamento, DA: *Data Augmentation*

Capítulo 5

Resultados

O objetivo deste estudo é realizar o reconhecimento de sinais da Libras, buscando uma metodologia que leve em consideração à sensibilidade ao sinalizador, tendo um modelo com capacidade de generalização e que represente um contexto real. Para isso duas abordagens foram estruturadas: uma utilizando o vídeo e outra a trajetória manual, como entrada do sistema de reconhecimento. A Figura 57 exemplifica cada etapa da metodologia implementada.

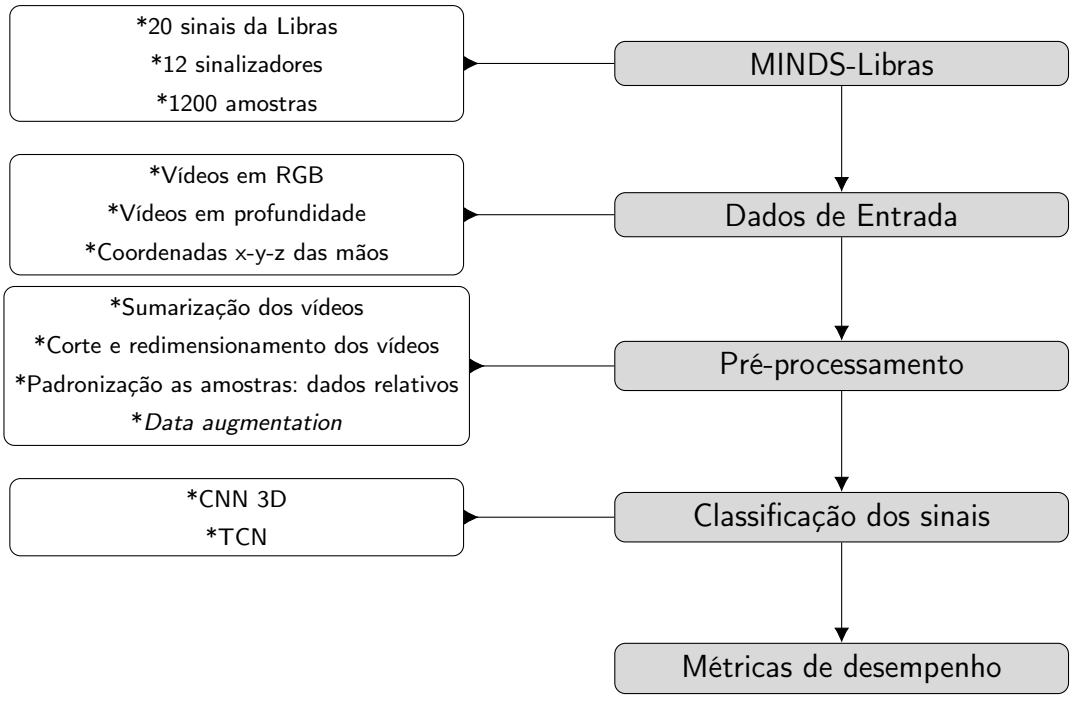


Figura 57 – Etapas das abordagens propostas nesta tese para o Reconhecimento Automático de Sinais da Libras.

A primeira abordagem foi estruturada com o intuito de explorar a metodologia apresentada em Rezende et al. (2021) e, com isso, realizar mudanças que permitissem um melhora no poder de generalização do modelo. Nesse caso a topologia da rede foi mantida e foram implementadas (i) a validação cruzada para separação dos dados de treino e teste;

(ii) a sumarização dos vídeos após corte dos quadros sem movimento; e (iii) a utilização de 5 quadros significativos para representar cada sinal. Na segunda abordagem, outra fonte de dados foi utilizada: pontos das mãos que caracterizam a trajetória manual durante a execução dos sinais. Nesse caso o conjunto de treino e teste foram separados por sinal e por sinalizador. Essa última opção teve o objetivo de testar a independência do modelo em relação ao sinalizador.

Para tratar e compilar todas essas informações foi necessária uma unidade de processamento do tipo GPU (Unidade de Processamento Gráfico) devido à sua capacidade em lidar com uma grande massa de dados. Entretanto, nem sempre é possível ter uma máquina física com tal unidade de processamento e a alternativa escolhida para compilar a metodologia estruturada nesta pesquisa foi desenvolver a implementação no ambiente *Google Colab Pro*. Com isso, as Seções 5.1 e 5.2 descrevem os resultados encontrados em cada abordagem, respectivamente. As métricas de avaliação utilizadas foram a acurácia média, precisão, *recall*, *F-measure* e curva ROC (*Receiver Operating Characteristic*). 5

5.1 Abordagem utilizando vídeos gravados em padrão RGB

Lidar com vídeos é uma tarefa custosa computacionalmente. No problema aqui tratado tem-se 1200 amostras, em que cada um passou pela etapa de sumarização, corte e redimensionamento, gerando um vídeo que foi representado por 5 imagens em uma resolução 224×224. Para a classificação, as amostras foram divididas em 12-*folds*, 10 para treinamento, 1 para validação e 1 para teste, e o procedimento repetido até que todos os *folds* passassem pela etapa de teste.

A análise desses resultados inicia com a Figura 58. Essa apresenta a matriz de confusão obtida após a classificação, utilizando a , de cada um dos 12-*folds*. Ela foi calculada e normalizada entre 0 e 1 para cada uma das iterações e obteve-se a média delas. Foram 1000 amostras de treinamento, 100 de validação e 100 de teste, alcançando uma taxa média de acerto de $84,75\%$. Estas iterações garantem uma aleatoriedade no conjunto de treino e de teste, fazendo com que ora a amostra participe do grupo de treinamento, ora do grupo de teste.

Para avaliar o modelo, a curva ROC foi obtida, como ilustra a Figura 59. A área abaixo da curva (*Area under the curve* - AUC) representa a probabilidade do modelo classificar corretamente os dados[1]. No caso desta abordagem o seu valor foi de $0,92$, indicando um ajuste satisfatório da curva ROC em um problema multiclasse.

O critério de comparação entre as abordagens realizadas neste trabalho foi a taxa

[1] A AUC próxima de 1 indica a capacidade do modelo em separar as classes.

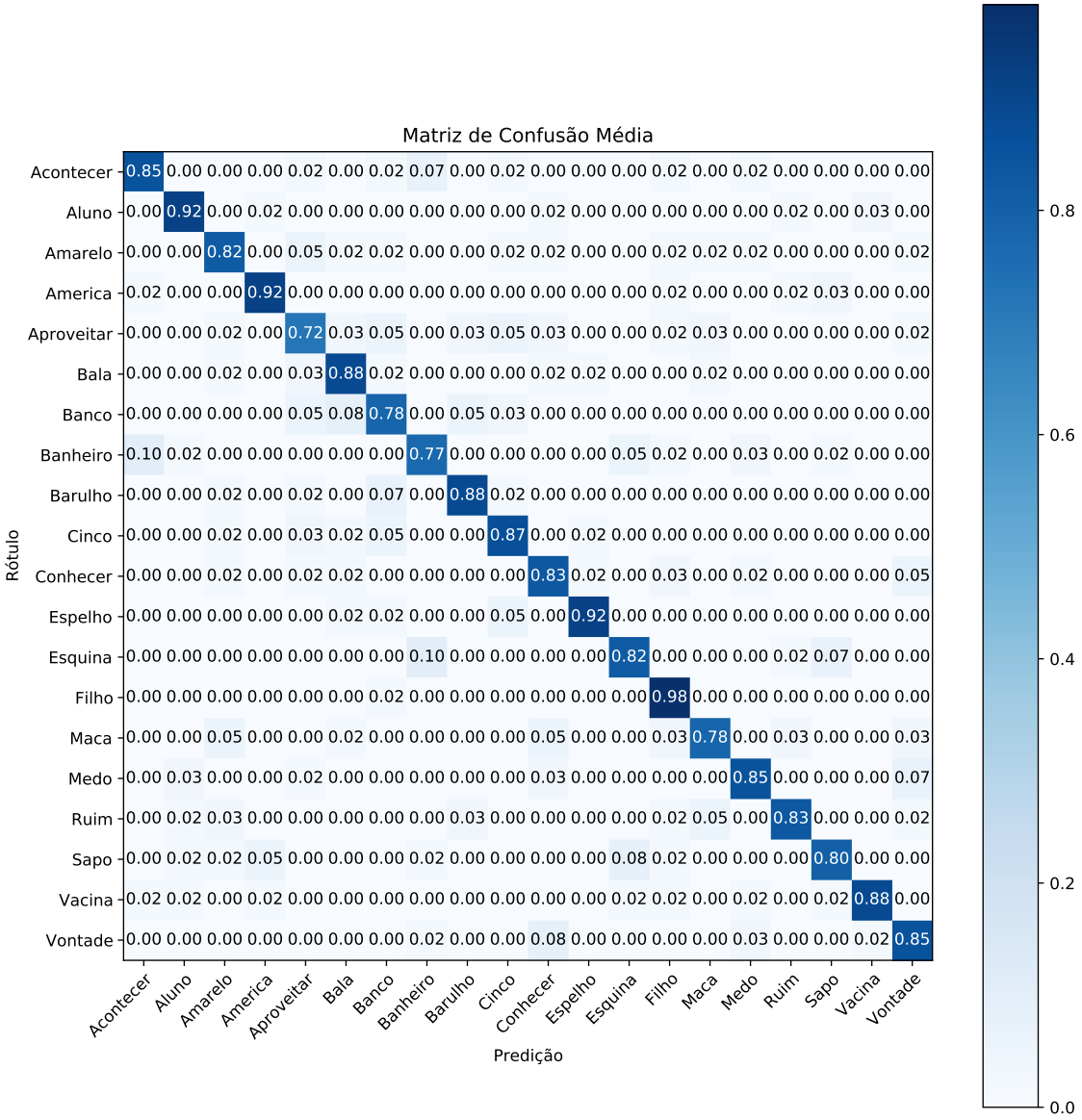


Figura 58 – Matriz de confusão normalizada obtida pela média 12-*folds*.

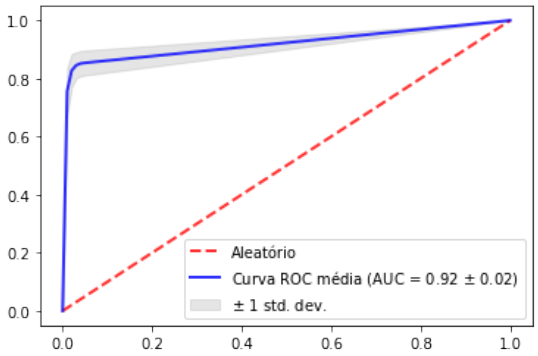


Figura 59 – Curva ROC da CNN 3D (AUC $= 0,92$).

média de acerto em relação ao conjunto de teste e a curva ROC. Entretanto, buscou-se métricas para avaliar cada classe em cada experimento. Dentre as várias presentes na literatura, optou-se pela *F-measure*, que considera o desempenho para a classe positiva (de Castro, 2011). Ela é calculada a partir de duas métricas: precisão (Equação 5.2) e *recall* (Equação 5.3), como mostra a Equação 5.1. De acordo com de Castro (2011), a variável β é utilizada para ajustar a importância relativa entre precisão e *recall* e pode ser tipicamente adotada como igual a 1.

$$F - measure = \frac{(1+\beta) * Recall * Precisao}{\beta^2 * Recall + Precisao} = \frac{2 * Recall * Precisao}{Recall + Precisao} \tag{5.1}$$

$$Precisao = \frac{VP}{VP + FP} \tag{5.2}$$

$$Recall = \frac{VP}{VP + FN} \tag{5.3}$$

sendo VP, verdadeiro positivo, o número de classificações corretas do sinal; FP, falso positivo, o número de amostras classificadas erroneamente como sendo da classe em questão e FN, falso negativo, o número de amostras da classe em questão classificadas erroneamente como sendo de outras classes.

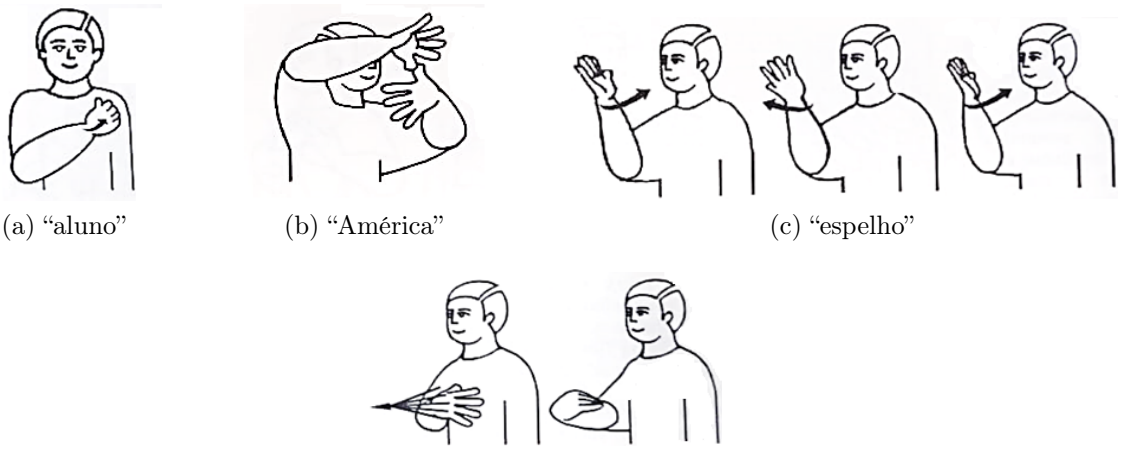
A acurácia média indica o desempenho geral do modelo, isto é, dentre todas as classificações, quantas o modelo classificou corretamente. Já a precisão e o *recall* foram realizados por classe. De acordo com Rodrigues (2019), a precisão pode ser usada em uma situação em que os FP's são considerados mais prejudiciais que os FN's e o *recall* pode ser usado em uma situação em que os FN's são considerados mais prejudiciais que os FP's. No caso deste trabalho, ambas as situações são importantes e não é interessante que a porcentagem de valores falsos seja elevada. Dessa forma, a *F-measure* com $\beta = 1$ realiza uma média harmônica entre estas duas métricas. De acordo com ela, quando tem-se um *F-measure* baixo é um indicativo de que ou a precisão ou o *recall* está baixo. A Tabela 11 apresenta as métricas calculadas por classe.

Entre os sinais que obtiveram as maiores taxas de acerto destacam-se "aluno" "América", "espelho" e "filho", ilustrados na Figura 60. O segundo é um sinal estático. Há, entretanto, um movimento dos braços para executá-lo. Já no caso dos demais, o ponto de articulação é similar quando comparado com outros sinais da base, mas o movimento é distinto. Nesses casos a porcentagem de amostras classificadas erroneamente foi pequena, abaixo de 5%, e distribuída entre poucos sinais.

Os sinais "banheiro" e "esquina" foram os mais confundidos pelo modelo. Em média, 10% das observações referentes ao sinal "banheiro" foram classificadas erroneamente

Tabela 11 – Métricas de desempenho da abordagem com a CNN 3D.

Sinal	***F-measure***	***Recall***	**Precisão**
Acontecer	0,85	0,83	0,86
Aluno	0,90	0,91	0,89
Amarelo	0,80	0,80	0,80
América	0,91	0,91	0,91
Aproveitar	**0,73**	**0,72**	0,75
Bala	0,84	0,87	0,81
Banco	0,76	0,79	**0,74**
Banheiro	0,77	0,76	0,79
Barulho	0,88	0,87	0,89
Cinco	0,84	0,86	0,82
Conhecer	0,79	0,82	0,77
Espelho	**0,92**	0,91	0,94
Esquina	0,82	0,81	0,84
Filho	0,90	**0,98**	0,84
Maçã	0,83	0,79	0,87
Medo	0,85	0,85	0,86
Ruim	0,86	0,83	0,90
Sapo	0,82	0,79	0,85
Vacina	**0,92**	0,86	**0,95**
Vontade	0,82	0,85	0,80



(a) “aluno” (b) “América” (c) “espelho”

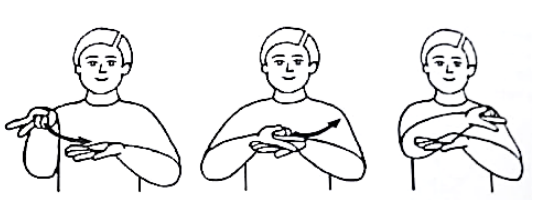
(d) “filho”

Figura 60 – Sinais que tiveram as maiores taxas de reconhecimento.
Fontes: Capovilla et al. (2012a, 2017a,b)

como “acontecer” e, com essa mesma proporção o sinal “esquina” foi confundido com o sinal “banheiro”. Todos eles são dinâmicos, como ilustra a Figura 61, mas, nesses casos, percebeu-se que ambos possuem ponto de articulação muito próximos e que podem ser bem similares dependendo da forma como o sinalizador executa o sinal. Além disso, em todos esses sinais, o posicionamento do braço esquerdo é semelhante.

O sinal com a menor taxa de acerto foi o “aproveitar”. As métricas de desempenho

(a) “banheiro”



(b) “acontecer”

(c) “esquina”

Figura 61 – Sinal “banheiro” classificado erroneamente como sinal “acontecer” e sinal “esquina” classificado erroneamente como sinal “banheiro”.
Fontes: Capovilla et al. (2012a,b, 2017a,b,c)

corroboraram com esse resultado e mostraram que 26% desse sinal foi classificado erroneamente (falsos negativos) e 24% de outras amostras foram rotuladas equivocadamente como tal (falsos positivos). Melhorar essas taxas de acerto pode ser possível analisando os parâmetros fonológicos para que a CNN não se especialize na localização do sinal. Entretanto, como os dados de entrada são imagens com uma sequência temporal, a rede acaba se especificando também nas características físicas do sinalizador, como ilustra a Figura 63. Essas imagens ilustram a saída da primeira camada convolucional (*Conv3D_1*), que possui 4 mapas de características, com 3 imagens de resolução 222×222. Além de destacar o sinalizador, a rede também evidenciou a mão que estava em movimento. Dessa forma, esse parâmetro tem grande impacto no reconhecimento do sinal e analisar a sua trajetória foi uma das motivações para a investigação da segunda abordagem, apresentada na Seção 4.2. Outro ponto que levou à próxima implementação foi a busca por um modelo que não identifique o sinalizador.

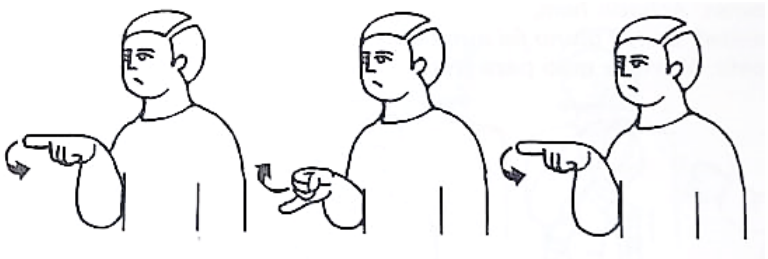


Figura 62 – Sinal “aproveitar” que apresentou pior desempenho nas métricas analisadas.
Fontes: Capovilla et al. (2012a,b, 2017a,b,c)

(a) 1º mapa de características

(b) 2º mapa de características

(c) 3º mapa de características

(d) 4º mapa de características

Figura 63 – Saída do primeiro bloco convolucional da CNN 3D proposta (*Conv3D_1*) na execução do sinal "banco": 3 imagens 222×222 para cada um dos 4 mapas de características.

5.2 Abordagem utilizando a informação das mãos

O primeiro teste realizado nesta abordagem foi a validação da análise descritiva apresentada na Seção 3.3.2. A importância dos 5 pontos de cada mão (mãos, pulsos, ponta das mãos, polegares e cotovelos) foi investigada utilizando a metodologia descrita na Seção 4.2, sem a implementação do *data augmentation*. O processo foi iniciado com os pontos relativos às mãos e os demais foram incluídos por etapas, conforme apresentado na Tabela 12. Concluiu-se que as coordenadas de todas essas juntas devem ser consideradas, mesmo que elas sejam próximas umas das outras e tenham trajetórias semelhantes, corroborando o estudo realizado previamente.

Tabela 12 – Análise dos pontos das mãos.

Juntas	Pontos	Acurácia média
Mãos	8 e 12	46.16%
↪ + Pulsos	↪ + 7 e 11	50.88%
↪ + Ponta das mãos	↪ + 22 e 24	55.88%
↪ + Polegares	↪ + 23 e 25	56.73%
↪ + Cotovelos	↪ + 6 e 10	67.76%

Em seguida o modelo da TCN foi aplicado às 30 séries temporais[2] que representam cada amostra da base de dados MINDS-Libras. O primeiro experimento realizado nesta abordagem teve como objetivo confrontar os resultados apresentados em Rezende et al. (2021) e na abordagem anterior, isto é, dividindo os dados por sinal. A matriz de entrada, composta pelas coordenadas x-y-z dos 10 pontos das mãos, foi classificada pela TCN, sendo o conjunto de treinamento composto pelas 1100 amostras originais mais 3300 geradas com as técnicas de *data augmentation* e o conjunto de teste mantinha as 100 amostras separadas inicialmente.

A primeira métrica de desempenho computada foi a matriz de confusão média normalizada. Ela foi obtida após 30 iterações do algoritmo de classificação utilizando a TCN, como apresenta a Figura 64. Foram 4400 amostras de treinamento e 100 de teste, alcançando uma taxa média de acerto de 96%. Para avaliar o modelo, a curva ROC foi obtida, como ilustra a Figura 65, com uma AUC de $0,98$. Esses dois valores são superiores à abordagem dos vídeos com a CNN 3D, representam um modelo com maior capacidade de generalização e que tem um tempo de treinamento reduzido em 23%. Todos as taxas de acerto tiveram melhoras significativas e, consequentemente, os falsos negativos também foram menores. Nesta abordagem, a maior confusão realizada foi em 8% das amostras do sinal "bala" que foram classificadas erroneamente como "cinco" e apenas dois sinais tiveram uma acurácia menor que 90%.

[2] Cada série temporal representa uma coordenada de cada um dos 10 pontos da mãos. Veja a Figura 53.

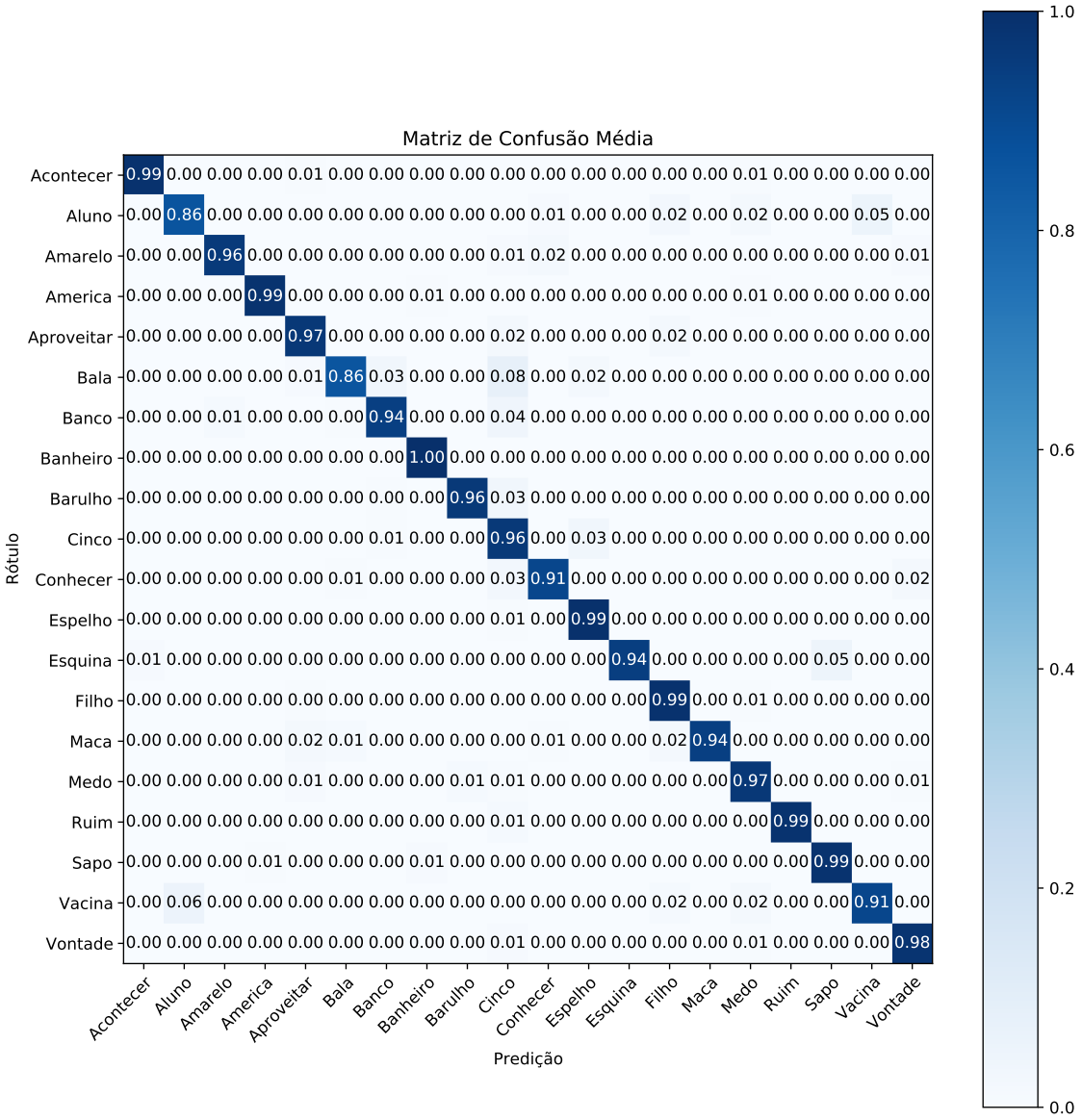


Figura 64 – Matriz de confusão normalizada obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinal.

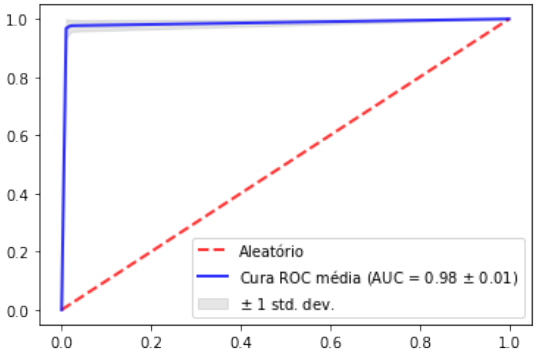


Figura 65 – Curva ROC da TCN quando os conjuntos de treino e teste foram divididos por sinal (AUC $= 0,98$).

Para a analisar o desempenho de cada classe, as métricas *F-measure*, *recall* e precisão foram calculadas e apresentadas na Tabela 13. Os sinais "esquina", "maçã" e "ruim" não computaram nenhum falso positivo, enquanto o sinal "banheiro" não foi confundido com nenhuma outra amostra.

Tabela 13 – Métricas de desempenho obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinal.

Sinal	***F-measure***	***Recall***	**Precisão**
Acontecer	0,98	0,98	0,99
Aluno	0,91	0,90	0,93
Amarelo	0,97	0,96	0,99
América	0,98	0,98	0,99
Aproveitar	0,95	0,96	0,95
Bala	0,92	**0,86**	0,98
Banco	0,95	0,95	0,96
Banheiro	**0,99**	**1,00**	0,98
Barulho	0,98	0,97	0,99
Cinco	**0,87**	0,96	**0,79**
Conhecer	0,95	0,94	0,96
Espelho	0,97	0,99	0,95
Esquina	0,97	0,94	**1,00**
Filho	0,96	0,99	0,93
Maçã	0,97	0,94	**1,00**
Medo	0,94	0,96	0,92
Ruim	**0,99**	0,99	**1,00**
Sapo	0,96	0,98	0,95
Vacina	0,92	0,90	0,95
Vontade	0,97	0,98	0,96

Os resultados apresentados propõem um novo *baseline* para a base da dados MINDS-Libras, no caso de uma metodologia baseada na divisão dos conjuntos de treino e teste por sinal. Este trabalho, entretanto, tem como foco a abordagem independente do sinalizador. Isso significa que a pessoa que participa do grupo de treino, não estará no grupo de teste e vice-versa. Essa abordagem é desafiadora pois o movimento realizado na execução de cada sinal, sua velocidade e amplitude tem variações de sinalizador para sinalizador. Essa característica está presente na língua de sinais e a alternativa adotada para minimizar essas diferenças foram as técnicas de *data augmentation*. Entretanto, como mostra a matriz de confusão da Figura 66, muitos sinais foram confundidos. Destacam-se nessa análise os 24% das amostras de "vacina" que foram confundidas com o sinal "aluno" e os 23% de "esquina" que foram classificadas erroneamente como sinal "sapo", representados nas Figuras 67 e 68. No primeiro caso, justifica-se a confusão pelo fato de ambos sinais terem o mesmo ponto de articulação e movimentos iniciais muito similares. Já para o segundo, o posicionamento dos braços e o local que o sinal é executado são os mesmos. Em relação aos demais resultados, percebeu-se a mesma tendência de erro sempre que os

sinais possuíam algum parâmetro fonológico similar. Em uma análise geral, o modelo teve uma taxa de acerto de $74,9\%$ e um tempo de treinamento inferior às abordagens por sinal já apresentadas.

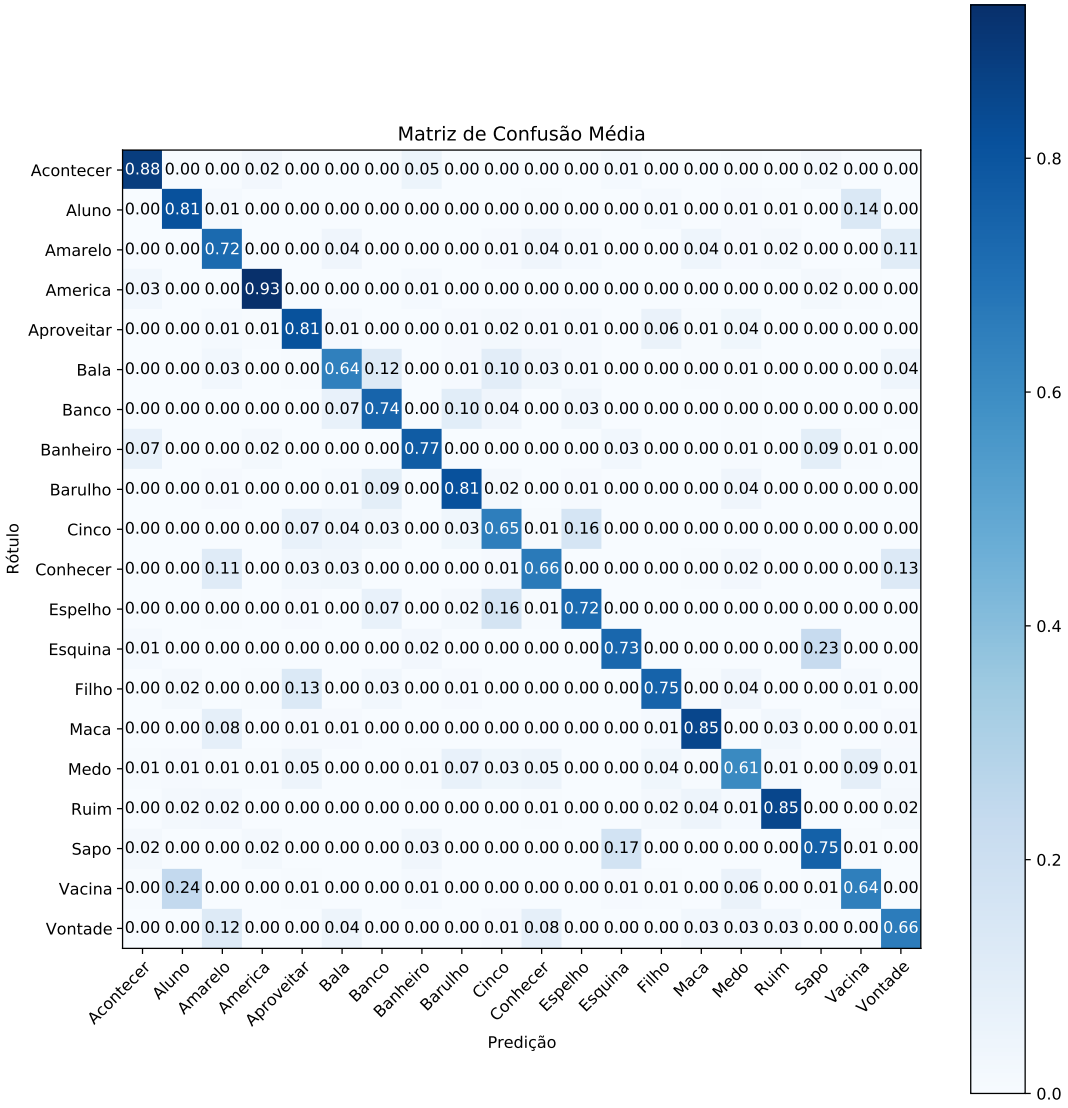
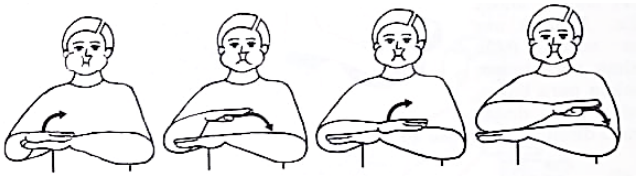


Figura 66 – Matriz de confusão normalizada obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinalizador.

(a) “vacina” (b) “aluno”

Figura 67 – Sinal “vacina” classificado erroneamente como sinal “aluno”. Fontes: Capovilla et al. (2012a,b, 2017a,b,c)

(a) “esquina”

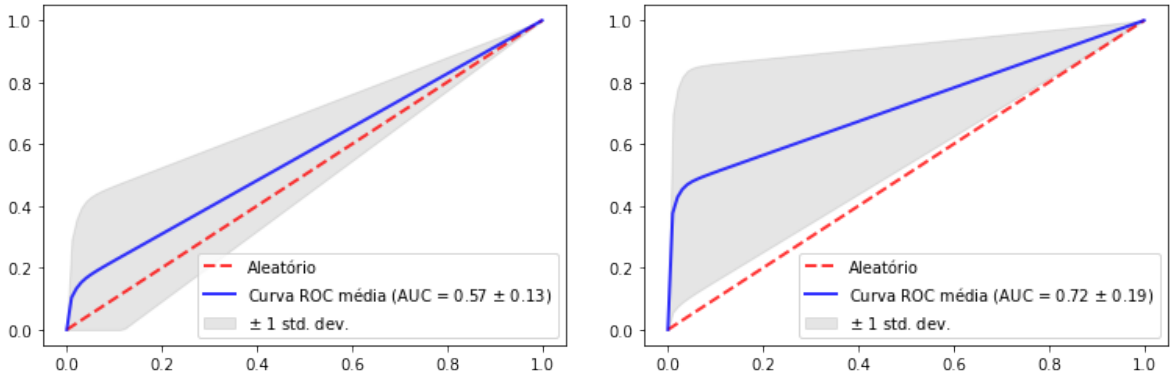


(b) “sapo”

Figura 68 – Sinal “esquina” classificado erroneamente como sinal “sapo”.
Fontes: Capovilla et al. (2012a,b, 2017a,b,c)

As metodologias com o foco em modelos independentes do sinalizador, seja no reconhecimento de gestos (Serrão et al., 2021) ou da língua de sinais (Sincan e Keles, 2020; Al-Hammadi et al., 2020a; Pan et al., 2020b), apresentaram resultados em que o desempenho cai significantemente com esta abordagem (*leave-one-subject-out*), como o ocorrido nesta pesquisa. Entretanto, o reconhecimento de sinais da Libras utilizando a trajetória manual juntamente com a TCN se torna atrativo pelo dados apresentados na Figura 69. Inicialmente 6 sinalizadores foram selecionados arbitrariamente para a classificação do sinal. Como mostra a Figura 69a, a curva ROC média se encontra próximo da aleatória e que ainda assim é um bom resultado considerando que este é um problema multiclasse. Além disso, a Figura 69b, que representa o resultado com todas as pessoas que compõem a MINDS-Libras, mostra que com mais indivíduos o modelo melhora a sua capacidade de generalização. Destaca-se a importância do número de sinalizadores que compõem a base de dados, permitindo que ela seja realmente representativa. Vale ressaltar que o aumento de dados de forma sintética (*data augmentation*), que é uma alternativa quando não se tem a possibilidade de aumentar a base, não resolve essa questão, porque captar todas as possíveis variações de execução dos sinalizador sem ter uma amostra exemplo é quase impossível.

A forma como cada sinalizador executa os sinais não é uma variável controlável. Essa é uma característica intrínseca de cada indivíduo e independente de qualquer padronização realizada durante a gravação da base de dados. A Tabela 14 apresenta o desempenho médio de cada sinalizador que participou da MINDS-Libras. Os resultados são reflexos do processo de gravação da base de dados que é exaustivo para os sinalizadores e, com isso, geram execuções com ruído. O maior valor de acurácia média foi obtida a partir dos testes com os dados de uma pessoa fluente na língua (Sinalizador 1), mostrando que conhecimento prévio e a vivência do idioma é um fator determinante para a padronização na execução dos sinais.



(a) 6 sinalizadores (AUC $= 0,57$) (b) 12 sinalizadores (AUC $= 0,72$)

Figura 69 – Curva ROC da TCN quando os conjuntos de treino e teste foram divididos por sinalizador.

Tabela 14 – Desempenho médio de cada sinalizador nas 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinalizador.

Sinalizadores	**Conhecimento prévio**	**Acurácia média**
1	Fluente (surdo)	**82,73** ± 8,22
2	Fluente (intérprete)	80,40 ± 8,98
3	Intermediário	65,50 ± 6,54
4	Fluente (professora)	76,20 ± 7,15
5	Intermediário	78,46 ± 5,89
6	Fluente (intérprete)	71,43 ± 7,33
7	Básico	78,16 ± 6,47
8	Básico	73,50 ± 8,96
9	Intermediário	79,63 ± 6,18
10	Intermediário	**60,73** ± 8,04
11	Intermediário	73,56 ± 7,93
12	Intermediário	79,46 ± 4,86

Por fim, as métricas de desempenho para sinal foram computadas, como ilustra a Tabela 15. Percebe-se que há um padrão entre os resultados apresentados nas outras abordagens, sendo que sinais com bons desempenho mantém valores de reconhecimento acima de 80%, enquanto os que resultavam em uma classificação equivocada na abordagem por sinal, acentuam essa diferença mesmo seguindo o padrão de gravação preestabelecido. Entretanto, vale ressaltar que, por se tratar de um problema multiclasse, com uma abordagem independente do sinalizador, os resultados cumpriram o objetivo deste estudo.

5.3 Principais Contribuições do Capítulo

Este capítulo apresentou o resultado da proposta metodológica, deixando claro o percurso para a obtenção dos resultados. As abordagens para classificação dos sinais tiveram como base a acurácia dos modelos nos conjuntos de teste, além de avaliar o seu

Tabela 15 – Métricas de desempenho obtida pela média 30 iterações da TCN quando os conjuntos de treino e teste foram divididos por sinalizador.

Sinal	***F-measure***	***Recall***	**Precisão**
Acontecer	0,88	0,90	0,86
Aluno	0,78	0,82	0,74
Amarelo	0,70	0,72	0,64
América	**0,93**	**0,94**	**0,92**
Aproveitar	0,76	0,81	0,72
Bala	0,68	0,65	0,72
Banco	0,72	0,76	0,69
Banheiro	0,82	0,77	0,87
Barulho	0,84	0,82	0,86
Cinco	**0,64**	0,66	**0,62**
Conhecer	0,70	0,67	0,73
Espelho	0,74	0,73	0,76
Esquina	0,75	0,74	0,77
Filho	0,79	0,76	0,83
Maçã	0,86	0,85	0,88
Medo	0,65	**0,61**	0,69
Ruim	0,87	0,86	0,89
Sapo	0,71	0,75	0,67
Vacina	0,67	0,64	0,71
Vontade	0,66	0,66	0,67

poder de generalização por meio da curva ROC e analisar cada sinal individualmente. O objetivo principal era que a rede aprendesse com os movimentos realizados pelas mãos, evitando a dependência do sinalizador. A Tabela 16 resume os experimentos realizados.

Novos *baselines* para a MINDS-Libras foram estabelecidos neste trabalho: o *Experimento* 2 para as abordagens por sinal e o *Experimento* 3 para metodologias independentes do sinalizador (*leave-one-signaller-out*).

Tabela 16 – Resultados dos experimentos realizados com a base de dados MINDS-Libras

Experimento	Entrada	Pré-processamento	Abordagem	Modelo	Acurácia média	AUC	Tempo (s)
Baseline	Escala de Cinza	S/R: 10 quadros e DA	Por sinal	CNN3D	93,30% (±1,69)	–	570,6
1	Escala de Cinza	S/R: 5 quadros	Por sinal	CNN3D	82,42% (±4,82)	0,91	78,75
	Escala de Cinza	↪ + DA	Por sinal	CNN3D	84,75% (±4,42)	0,92	262,1
2	*Position* x-y-z	DA	**Por sinal**	TCN	**96,00% (±6,69)**	0,98	136,0
3	*Position* x-y-z	DA / 6 sinalizadores	Por sinalizador	TCN	62,40% (±9,71)	0,57	34,32
	Position x-y-z	DA / 12 sinalizadores	**Por sinalizador**	TCN	**74,90% (±9,52)**	0,72	65,24

*S/R: Sumarização e Redimensionamento, DA: *Data Augmentation*

Capítulo 6

Conclusões

O reconhecimento é uma habilidade cognitiva que o cérebro adquire para identificar pessoas, lugares ou coisas memorizadas no passado. É uma capacidade que vai sendo aprimorada à medida que o número de repetições de cada situação se amplifica. Nesse contexto, a Inteligência Computacional surgiu com o intuito de desenvolver esse tipo de habilidade nas máquinas. Para isso, vários campos de pesquisa são explorados, seja o Processamento de Imagens, a Visão Computacional, as Redes Neurais Artificias e o Aprendizado Profundo, e em diversos momentos esses campos se entrelaçam.

Ainda no âmbito das aptidões estão as formas de comunicação que utilizam diversos meios para se propagar. A língua de sinais é uma delas, sendo uma forma de transmissão de ideias que utiliza a configuração e movimento das mãos, a orientação da palma da mão, o espeço que o sinal é executado e as expressões não-manuais como unidades formacionais. Identificar um sinal computacionalmente é interpretar esses parâmetros fonológicos.

Com isso, este trabalho propôs realizar o reconhecimento automático de sinais da Libras utilizando abordagens baseadas em Redes Neurais Convolucionais. Para a estruturação dos experimentos, uma revisão dos artigos que trabalharam com o reconhecimento da língua de sinais foi realizada para que fosse possível apresentar o desenvolvimento do reconhecimento computacional da Libras frente às demais línguas de sinais, além de destacar o contexto histórico e social em que a área se encontra. Em seguida, a base de dados MINDS-Libras foi criada e disponibilizada publicamente para que fosse possível validar a metodologia proposta. Dentre os dados que a compõem, este trabalho investigou duas estratégias: (i) uma com os vídeos em RGB e a Rede Neural Convolucional 3D (CNN 3D), e (ii) outra com a trajetória manual e a Rede Neural Convolucional Temporal (TCN). A metodologia em ambos os casos foi a mesma, se diferenciando e adaptando algumas etapas à fonte de dados que foi utilizada.

A hipótese investigada neste trabalho foi que a melhor metodologia para reconhecer automaticamente os sinais, independente do sinalizador (*leave-one-signaller-out*), seria considerando a informação temporal do movimento e a trajetória das mãos. Os experimentos

realizados mostraram que CNN 3D apresentou resultados promissores quando o conjunto de treino e teste foram divididos por sinal. Entretanto, as saídas das camadas convolucionais mostraram que essa abordagem se especifica nas características físicas do sinalizador. Dessa forma, a metodologia utilizando o movimento manual e a TCN tornou-se promissora tanto na divisão dos dados por sinal, quanto por sinalizador.

Por fim, vale ressaltar que os resultados obtidos neste estudo são frutos de pesquisas desenvolvidas desde 2014: Almeida (2014b); Almeida et al. (2014); Rezende et al. (2016b); Rezende (2016); Rezende et al. (2016a, 2017); Guerra et al. (2018); Assis (2018); Guerra (2019); Castro et al. (2019); Castro (2020); Rezende et al. (2021). Cada trabalho apresenta uma abordagem específica, investiga diferentes dados e metodologias para a classificação de sinais da Libras. Entretanto, esses estudos representam uma evolução do grupo de pesquisa e serviram como base para a estrutura proposta neste trabalho.

Para complementar a conclusão deste estudo, a Seção 6.1 apresenta uma análise crítica dos modelos desenvolvidos. Em seguida, na Seção 6.2, são propostos caminhos promissores para o reconhecimento de sinais da Libras e, na Seção 6.3, são expostos os trabalhos publicados durante a elaboração desta tese.

6.1 Abordagens Propostas

O primeiro desafio desta pesquisa foi a criação da base de sinais da Libras. A escolha dos sinais, sinalizadores e dos sensores não foi uma tarefa trivial dada a gama de possibilidade que se tem acesso. Além disso, vale ressaltar que o processo de gravação era exaustivo para os sinalizadores e o seu tempo também foi levado em consideração. Os elementos utilizados não torna rígida a reprodução da base e com o protocolo de gravação definido previamente torna-se possível sua reprodutibilidade e escalonabilidade.

A MINDS-Libras possui 20 sinais gravados 5 vezes por 12 sinalizadores (= 1200 amostras) e está disponibilizada publicamente em Minds (2019). Ela é limitada quando se diz respeito ao número de sinais e sinalizadores, e os experimentos realizados mostram que apesar do protocolo de gravação ter sido bem documentado, uma variação maior das características físicas dos indivíduos possibilitaria um aprendizado mais representativo. Se houvesse um conjunto de dados mais robusto, não seria necessário, por exemplo, aplicar o *data augmentation* e, com isso, eliminaria-se um pré-processamento que torna a metodologia mais custosa. Isso significa que o desempenho do sistema está diretamente relacionado aos dados que se tem, deixando explícito que é desejável que os sinais sejam executados pelo maior número de pessoas possível, tornando a base representativa.

A gravação dos dados dessa base foi realizada por dois sensores: (i) uma câmera RGB (Canon EOS Rebel t5i) e (ii) um sensor RGB-D (*Kinect v2* para *Xbox One*). Como as informações relativas à câmera foram massivamente utilizadas em Assis (2018), Guerra

(2019), Castro et al. (2019), Castro (2020) e Rezende et al. (2021), este trabalho investigou a contribuição dos dados disponibilizados pelo sensor RGB-D: (i) vídeos em RGB com resolução 1920×1080, (ii) vídeos da informação de profundidade em 640×480 *pixels*, (iii) informações de 25 juntas do corpo ao longo dos 150 quadros que compõem as gravações e de (iv) 1347 pontos da face do sinalizador. Como o foco do trabalho foi na trajetória manual, os pontos da face não foram abordados neste trabalho e os dados de profundidade foram considerados, mas não trouxeram melhoras nas métricas de desempenho dos modelos.

A validação da MINDS-Libras foi realizada utilizando duas diretrizes. A primeira, com os vídeos e a CNN 3D, buscou refinar o que foi apresentado em Rezende et al. (2021) de forma que os resultados encontrados representassem a real capacidade de generalização do modelo. Uma das etapas dessa metodologia que necessita de ajustes é a determinação do número de quadros significativos de cada amostra. Não há um padrão em relação a esse número e uma má escolha dele pode gerar erros nas etapas posteriores do sistema de classificação. Ainda assim, o pré-processamento realizado refinou a sumarização e retornou as mostras mais representativas quando comparado com o artigo base. Já a segunda diretriz, que investigou a trajetória manual e a TCN, mostrou resultados melhores nas métricas de desempenho e uma metodologia com um tempo de processamento inferior ao anterior. Como abordado anteriormente, o *data augmentation* foi um pré-processamento que poderia ter sido evitado, se houvesse uma base de dados mais robusta. Necessita-se de amostras em grande quantidade pra fazer o estudo de forma satisfatória.

Para a avaliação das metodologias propostas foram considerados várias métricas: acurácia, *f-measure*, precisão, *recall*, tempo de treinamento, curva ROC e AUC. Como apresentaram os resultados, não necessariamente uma taxa de acerto acima de 90%, por exemplo, significa que o modelo consegue ser generalista. A forma de modelar o experimento impacta diretamente nos resultados e este trabalho buscou analisar cautelosamente cada parâmetro.

A potencialidade da metodologia proposta nessa tese se comprova analisando os estudos presentes na literatura. Os trabalhos que realizaram o reconhecimento automático da língua de sinais mostraram que há uma tendência quando se diz respeito (i) às unidades fonológicas utilizadas nas metodologias de classificação e (ii) à aquisição dos dados. Nesses casos, os parâmetros manuais e as técnicas de Visão Computacional foram os elementos que se destacaram no contexto histórico da área. Entretanto, lidar com dados que representam uma série temporal, como na abordagem da TCN, é voltar nas técnicas com sensores "vestíveis" que lidam com dados dessa natureza e que foram consideradas inviáveis devido ao aparato eletrônico que deve ser acoplado na mão/braço do sinalizador. Contudo, considerando a potencialidade das metodologias que utilizam dados com variação temporal e que nem sempre tem-se disponível um sensor que captura os pontos do corpo, a literatura apresenta técnicas de *Human Pose Estimation* que permitem estimar tais juntas, tornando

a abordagem com a TCN ainda mais robusta e independente do sensor que capture pontos do corpo.

Por fim, acredita-se que, mesmo com as lacunas da área de reconhecimento automático de sinais apresentadas na revisão bibliográfica (Capítulo 2) e que precisam ser solucionados na Libras, a base de dados implementada contribuirá de forma expressiva para o desenvolvimento de novas aplicações no setor e temas correlatos, e a metodologia estruturada servirá como referência para trabalhos que utilizarão o mesmo conjunto de dados e sistemas que realizarem o reconhecimento automático da Libras.

6.2 Futuras Investigações

Tendo em vista os objetivos desta pesquisa e o contexto em que a área de reconhecimento automático de sinais da Libras se encontra, as seguintes atividades são sugeridas para trabalhos futuros:

1. Fomentar a base de dados MINDS-Libras com o foco na diversificação de sinalizadores e de sinais da língua brasileira. Lembrando que há muitos sinais que ainda não estão documentados, como os que se referem às mais diversas áreas técnicas e, nesses casos, eles ainda são realizados por meio da datilologia[1];

2. Realizar uma análise quantitativa da MINDS-Libras, buscando correlação entre as amostras;

3. Aprimorar a técnica de sumarização, obtendo o número de quadros significativos a partir de um algoritmo que será aplicado a um conjunto de dados tomado como referência inicial;

4. Explorar os demais parâmetros fonológicos da língua de sinais, buscando características que distinguam os sinais. Uma metodologia de reconhecimento que se especifique em cada unidade formacional individualmente, pode concatenar as informações ao final do processo, tornando a abordagem preparada para receber novas amostras e o modelo mais generalista;

5. Expandir as técnicas de *data augmentation* aplicadas nas abordagens com a CNN 3D e aumentar a variação dos parâmetros das redes;

6. Investigar o desempenho da abordagem com a TCN para o problema de Aprendizado de Máquina *Seq-to-Seq* (sequência para sequência), utilizado no processamento de linguagens na tradução de idiomas. Neste caso, uma sequência é transformada em outra de mesmo significado, mas de naturezas diferentes. Confrontar os resultados com as RNN's e LSTM's comumente utilizadas para esse tipo de problema;

[1] A datilologia é similar a soletração na língua oral: representação de cada letra.

7. Averiguar uma abordagem hierárquica com mecanismo de atenção, combinando as informações disponibilizadas pela MINDS-Libras.

6.3 Publicações

Os seguintes trabalhos científicos foram aceitos para publicação durante a elaboração desta tese e suas versões completas estão expostas no Apêndice C. Neste mesmo período, outros trabalhos foram desenvolvidos, focando na divulgação desta pesquisa e em assuntos complementares. O Apêndice C.4 sumariza todos eles.

- Guerra, R. R., **Rezende, T. M.**, Guimaraes, F. G., & Almeida, S. G. M. Facial Expression Analysis in Brazilian Sign Language for Sign Recognition. *In Anais do XV Encontro Nacional de Inteligência Artificial e Computacional* (pp. 216-227). SBC, 2018;

- Almeida, Sílvia G. M., **Rezende, Tamires M.**, Almeida, Gabriela T. B., Toffolo, Andreia C. R, & Guimarães, Frederico G. (2019). MINDS-Libras Dataset [Data set]. Zenodo. http://doi.org/10.5281/zenodo.2667329;

- de Castro, G. Z., Guerra, R. R., de Assis, M. M., **Rezende, T. M.**, de Almeida, G. T., Almeida, S. G., & Guimaraes, F. G. Desenvolvimento de uma Base de Dados de Sinais de Libras para Aprendizado de Máquina: Estudo de Caso com CNN 3D. *In Anais do XIV Simpósio Brasileiro de Automação Inteligente* (pp. 216-227). SBA, 2019;

- Almeida, Sílvia G. M., **Rezende, Tamires M.**, Almeida, Gabriela T. B., Toffolo, Andreia C. R., & Guimarães, Frederico G. (2020). MINDS-Libras Dataset (RGB-D sensor data) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.4322984;

- **Rezende, T. M.**, Almeida, S. G. M., & Guimarães, F. G. (2021). Development and validation of a Brazilian sign language database for human gesture recognition. *Neural Computing and Applications*, 1-19.

Referências

V. Adithya e R. Rajesh. Hand gestures for emergency situations: A video dataset based on words from indian sign language. *Data in Brief*, 31:106016, 2020a. ISSN 2352-3409. doi: https://doi.org/10.1016/j.dib.2020.106016. URL https://www.sciencedirect.com/science/article/pii/S2352340920309100.

V. Adithya e R. Rajesh. A deep convolutional neural network approach for static hand gesture recognition. *Procedia Computer Science*, 171:2353–2361, 2020b.

H. H. Aghdam e E. J. Heravi. Guide to convolutional neural networks. *New York, NY: Springer. doi*, 10:978–3, 2017.

U. V. Agris. Database for Signer-Independent Continuous Sign Language Recognition, 2008. URL https://www.phonetik.uni-muenchen.de/forschung/Bas/SIGNUM/.

H. F. T. Ahmed, H. Ahmad, K. Narasingamurthi, H. Harkat, e S. K. Phang. Df-wislr: Device-free wi-fi-based sign language recognition. *Pervasive and Mobile Computing*, 69: 101289, 2020.

M. Ahmed, B. Zaidan, A. Zaidan, M. M. Salih, Z. Al-qaysi, e A. Alamoodi. Based on wearable sensory device in 3d-printed humanoid: A new real-time sign language recognition system. *Measurement*, 168:108431, 2021.

M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, T. S. Alrayes, H. Mathkour, e M. A. Mekhtiche. Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. *IEEE Access*, 8: 192527–192542, 2020a.

M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, e M. A. Mekhtiche. Hand gesture recognition for sign language using 3dcnn. *IEEE Access*, 8: 79491–79509, 2020b.

M. Al-Rousan, K. Assaleh, e A. Tala'a. Video-based signer-independent arabic sign language recognition using hidden markov models. *Applied Soft Computing*, 9(3): 990–999, 2009.

G. T. B. Almeida. Criação de banco de sinais de libras para implementação de sistemas com visão computacional. Trabalho de Conclusão de Curso, 2017.

S. G. M. Almeida. Libras-34 Dataset (Kinect v1), 2014a. URL http://doi.org/10.5281/zenodo.4451526. Accessed on 09/03/2020.

S. G. M. Almeida. *Extração de Características em Reconhecimento de Parâmetros Fonológicos da Língua Brasileira de Sinais utilizando Sensores RGB-D*. PhD thesis, Universidade Federal de Minas Gerais, Programa de Pós Graduação em Engenharia Elétrica, Belo Horizonte, Minas Gerais, Brasil, 2014b.

S. G. M. Almeida, F. G. Guimarães, e J. A. Ramírez. Feature extraction in brazilian sign language recognition based on phonological structure and using rgb-d sensors. *Expert Systems with Applications*, 41(16):7259–7271, 2014.

S. G. M. Almeida, A. R. R. Freitas, e F. G. Guimarães. Um método para sumarização de vídeos baseado no problema da diversidade máxima e em algoritmos evolucionários. In *XII Simpósio Brasileiro de Automação Inteligente (SBAI)*, pages 1298 – 1303, Natal, Rio Grande do Norte, Brasil, 2015.

S. G. M. Almeida, T. M. Rezende, A. C. R. Toffolo, e C. L. Castro. Libras-10 dataset, 2016. URL http://doi.org/10.5281/zenodo.3229958.

S. Aly e W. Aly. Deeparslr: A novel signer-independent deep learning framework for isolated arabic sign language gestures recognition. *IEEE Access*, 8:83199–83212, 2020.

O. Aran, I. Ari, L. Akarun, B. Sankur, A. Benoit, A. Caplier, P. Campr, A. H. Carrillo, et al. Signtutor: An interactive system for sign language tutoring. *IEEE MultiMedia*, pages 81–93, 2009a.

O. Aran, T. Burger, A. Caplier, e L. Akarun. A belief-based sequential fusion approach for fusing manual signs and non-manual signals. *Pattern Recognition*, 42(5):812–822, 2009b.

M. Asadi-Aghbolaghi, A. Clapés, M. Bellantonio, H. J. Escalante, V. Ponce-López, X. Baró, I. Guyon, S. Kasaei, e S. Escalera. Deep learning for action and gesture recognition in image sequences: A survey. In *Gesture Recognition*, pages 539–578. Springer, 2017.

K. Assaleh e M. Al-Rousan. Recognition of arabic sign language alphabet using polynomial classifiers. *EURASIP Journal on Advances in Signal Processing*, 2005(13):507614, 2005.

M. M. d. Assis. Aplicação de técnicas de inteligência computacional para reconhecimento de sinais de libras. Trabalho Final de Curso, UFMG, 2018.

P. Athira, C. Sruthi, e A. Lijiya. A signer independent sign language recognition with co-articulation elimination from live videos: An indian scenario. *Journal of King Saud University-Computer and Information Sciences*, 2019.

V. Athitsos, C. Neidle, e S. Sclaroff. American sign language lexicon video dataset (asllvd), 2008. URL http://vlm1.uta.edu/~athitsos/asl_lexicon/.

D. Avola, M. Bernardi, L. Cinque, G. L. Foresti, e C. Massaroni. Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures. *IEEE Transactions on Multimedia*, 21(1):234–245, 2018.

S. G. Azar e H. Seyedarabi. Trajectory-based recognition of dynamic Persian sign language using hidden Markov model. *Computer Speech & Language*, 61:101053, 2020.

S. Bai, J. Zico Kolter, e V. Koltun. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *CoRR*, abs/1803.0, 2018. URL http://arxiv.org/abs/1803.01271.

F. I. Bashir, A. A. Khokhar, e D. Schonfeld. View-invariant motion trajectory-based activity classification and recognition. *Multimedia Systems*, 12(1):45–54, 2006.

R. Battison. Phonological deletion in american sign language. *Sign language studies*, 5(1): 1–19, 1974.

R. Battison. *Lexical borrowing in American sign language*. ERIC, 1978.

A. Ben Tamou, L. Ballihi, e D. Aboutajdine. Automatic learning of articulated skeletons based on mean of 3d joints for efficient action recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 31(04):1750008, 2017.

M. A. Bencherif, M. Algabri, M. A. Mekhtiche, M. Faisal, M. Alsulaiman, H. Mathkour, M. Al-Hammadi, e H. Ghaleb. Arabic sign language recognition system using 2d hands and body skeleton data. *IEEE Access*, 9:59612–59627, 2021.

V. Bloom, V. Argyriou, e D. Makris. Hierarchical transfer learning for online recognition of compound actions. *Computer Vision and Image Understanding*, 144:62–72, 2016.

M. Boulares e M. Jemni. 3d motion trajectory analysis approach to improve sign language 3d-based content recognition. *Procedia Computer Science*, 13:133–143, 2012.

Brasil. Lei n^o 10.436, de 24 de abril de 2002. *Diário Oficial [da] República Federativa do Brasil*, 2002. URL http://www.planalto.gov.br/ccivil_03/leis/2002/l10436.htm.

Brasil. Decreto n^o 5.626, de 22 de dezembro de 2005. *Diário Oficial [da] República Federativa do Brasil*, 2005. URL http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2005/decreto/d5626.htm.

T. D. Bui e L. T. Nguyen. Recognizing postures in vietnamese sign language with mems accelerometers. *IEEE sensors journal*, 7(5):707–712, 2007.

L. F. Cambuim, R. M. Macieira, F. M. Neto, E. Barros, T. B. Ludermir, e C. Zanchettin. An efficient static gesture recognizer embedded system based on elm pattern recognition algorithm. *Journal of Systems Architecture*, 68:1–16, 2016.

F. C. Capovilla e W. D. Raphael. *Enciclopédia da língua de sinais brasileiras: o mundo do surdo em libras*, volume 8. Edusp, 2004.

F. C. Capovilla, W. D. Raphael, e A. C. L. Maurício. *Dicionário Enciclopédico Ilustrado Trilíngue da Língua Brasileira de Sinais (Libras) baseado em Linguística e Neurociências Cognitivas, Volume I: Sinais de A a H.*, volume 1. Edusp, Brasil, 2 edition, 2012a. ISBN 9788531413315.

F. C. Capovilla, W. D. Raphael, e A. C. L. Maurício. *Dicionário Enciclopédico Ilustrado Trilíngue da Língua Brasileira de Sinais (Libras) baseado em Linguística e Neurociências Cognitivas, Volume II: Sinais de I a Z.*, volume 2. Edusp, Brasil, 2 edition, 2012b. ISBN 9788531413315.

F. C. Capovilla, W. D. Raphael, J. G. Temoteo, e A. C. Martins. *Dicionário da Língua Brasileira do Brasil: A Libras em suas mãos, Volume I: Sinais de A a D.*, volume 1. Edusp, São Paulo, Brasil, 1 edition, 2017a. ISBN 9788531415401.

F. C. Capovilla, W. D. Raphael, J. G. Temoteo, e A. C. Martins. *Dicionário da Língua Brasileira do Brasil: A Libras em suas mãos, Volume I: Sinais de A a D.*, volume 2. Edusp, São Paulo, Brasil, 1 edition, 2017b. ISBN 9788531415418.

F. C. Capovilla, W. D. Raphael, J. G. Temoteo, e A. C. Martins. *Dicionário da Língua Brasileira do Brasil: A Libras em suas mãos, Volume I: Sinais de P a Z.*, volume 3. Edusp, São Paulo, Brasil, 1 edition, 2017c. ISBN 9788531415425.

E. E. Cardenas e G. C. Chavez. Multimodal Hand Gesture Recognition Combining Temporal and Pose Information Based on CNN Descriptors and Histogram of Cumulative Magnitudes. *Journal of Visual Communication and Image Representation*, page 102772, 2020.

G. Caridakis, S. Asteriadis, e K. Karpouzis. Non-manual cues in automatic sign language recognition. *Personal and ubiquitous computing*, 18(1):37–46, 2014.

N. K. Caselli, Z. S. Sehyr, A. M. Cohen-Goldberg, e K. Emmorey. Asl-lex: A lexical database of american sign language, 2017. URL http://asl-lex.org/.

G. Z. Castro. *Reconhecimento de Linguas de Sinais Utilizando Redes Neurais Convolucionais e Transferencia de Aprendizado*. PhD thesis, Universidade Federal de Minas Gerais, Belo Horizonte, 2020.

G. Z. Castro, R. R. Guerra, M. M. Assis, T. M. Rezende, G. T. B. Almeida, S. G. M. Almeida, C. L. Castro, e F. G. Guimarães. Desenvolvimento de uma base de dados de sinais de libras para aprendizado de máquina: Estudo de caso com cnn 3d. In *XIV Simpósio Brasileiro de Automação Inteligente*, 2019.

L. R. Cerna, E. E. Cardenas, D. G. Miranda, D. Menotti, e G. Camara-Chavez. A multimodal LIBRAS-UFOP Brazilian sign language dataset of minimal pairs using a microsoft Kinect sensor. *Expert Systems with Applications*, 167, 4 2021. ISSN 09574174. doi: 10.1016/j.eswa.2020.114179.

F.-S. Chen, C.-M. Fu, e C.-L. Huang. Hand gesture recognition using a real-time tracking method and hidden markov models. *Image and vision computing*, 21(8):745–758, 2003.

C. Conly, P. Doliotis, P. Jangyodsuk, R. Alonzo, e V. Athitsos. Toward a 3d body part detection video dataset and hand tracking benchmark. In *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments*, page 2. ACM, 2013.

R. Cui, H. Liu, e C. Zhang. A deep neural framework for continuous sign language recognition by iterative training. *IEEE Transactions on Multimedia*, 21(7):1880–1891, 2019.

Y. Cui e J. Weng. Appearance-based hand sign recognition from intensity image sequences. *Computer Vision and Image Understanding*, 78(2):157–176, 2000.

J. Darby, M. B. Sánchez, P. B. Butler, e I. D. Loram. An evaluation of 3d head pose estimation using the microsoft kinect v2. *Gait & posture*, 48:83–88, 2016.

C. L. de Castro. *Novos critérios para seleçao de modelos neurais em problemas de classificaçao com dados desbalanceados*. PhD thesis, Universidade Federal de Minas Gerais, 2011.

A. Dertat. Applied deep learning - part 4: Convolutional neural networks, 2017. URL https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2.

B. Dorner e E. Hagen. Towards an american sign language interface. In *Integration of Natural Language and Vision Processing*, pages 143–161. Springer, 1994.

P. Dreuw, D. Rybach, T. Deselaers, M. Zahedi, e H. Ney. Rwth-boston-104 database, 2007. URL http://www-i6.informatik.rwth-aachen.de/~dreuw/database-rwth-boston-104.php.

R. Elakkiya e K. Selvamani. Extricating manual and non-manual features for subunit level medical sign modelling in automatic sign language classification and recognition. *Journal of Medical Systems*, 41(11):175, 2017.

R. Elakkiya e K. Selvamani. Enhanced dynamic programming approach for subunit modelling to handle segmentation and recognition ambiguities in sign language. *Journal of Parallel and Distributed Computing*, 117:246–255, 2018.

R. Elakkiya e K. Selvamani. Subunit sign modeling framework for continuous sign language recognition. *Computers & Electrical Engineering*, 74:379–390, 2019.

R. Elakkiya, P. Vijayakumar, e N. Kumar. An optimized generative adversarial network based continuous sign language classification. *Expert Systems with Applications*, page 115276, 2021.

A. S. Elons, M. Abull-ela, e M. F. Tolba. Neutralizing lighting non-homogeneity and background size in pcnn image signature for arabic sign language recognition. *Neural Computing and Applications*, 22(1):47–53, 2013.

S. Escalera, J. Gonzàlez, X. Baró, M. Reyes, O. Lopes, I. Guyon, V. Athitsos, e H. Escalante. Multi-modal gesture recognition challenge 2013: Dataset and results. In *Proceedings of the 15th ACM on International conference on In the interaction*, pages 445–452. ACM, 2013.

S. Escalera, V. Athitsos, e I. Guyon. Challenges in multi-modal gesture recognition. In *Gesture Recognition*, pages 1–60. Springer, 2017.

M. Fagiani, E. Principi, S. Squartini, e F. Piazza. A new Italian sign language database. In *International Conference on Brain Inspired Cognitive Systems*, pages 164–173. Springer, 2012.

M. Fagiani, E. Principi, S. Squartini, e F. Piazza. Signer independent isolated italian sign recognition based on hidden markov models. *Pattern Analysis and Applications*, 18(2): 385–402, 2015.

G. Fang, W. Gao, e D. Zhao. Large vocabulary sign language recognition based on fuzzy decision trees. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 34(3):305–314, 2004.

G. Fang, W. Gao, e D. Zhao. Large-vocabulary continuous sign language recognition based on transition-movement models. *IEEE transactions on systems, man, and cybernetics-part a: systems and humans*, 37(1):1–9, 2006.

T. A. Felipe. *Libras em Contexto: Curso Básico: Livro do Estudante*. WalPrint Gráfica Editora, Rio de Janeiro, 9 edition, 2009. ISBN 85-99091-01-8.

L. Fereira-Brito. Uma abordagem fonológica dos sinais da lscb. *Informativo Técnico-Científico do INES, Rio de Janeiro*, 1(1):20–43, 1990.

P. M. Ferreira, J. S. Cardoso, e A. Rebelo. On the role of multimodal learning in the recognition of sign language. *Multimedia Tools and Applications*, 78(8):10035–10056, 2019.

L. Ferreira-Brito. *Integração social e educação de surdos*. Babel Editora, 1993.

L. Ferreira-Brito. *Por uma gramática das línguas de sinais*. Tempo Brasileiro, Rio de Janeiro, 1995.

J. Forster, C. Schmidt, T. Hoyoux, O. Koller, U. Zelle, J. H. Piater, e H. Ney. Rwth-phoenix-weather, 2012. URL http://www-i6.informatik.rwth-aachen.de/~forster/database-rwth-phoenix.php.

A. R. R. Freitas, F. G. Guimarães, R. C. P. Silva, e M. J. F. Souza. Memetic self-adaptive evolution strategies applied to the maximum diversity problem. *Optimization Letters*, 8 (2):705–714, 2014a.

F. A. Freitas, F. V. Barbosa, e S. M. Peres. Grammatical Facial Expressions Data Set, 2014b. URL https://archive.ics.uci.edu/ml/datasets/Grammatical+Facial+Expressions.

F. d. A. Freitas. *Reconhecimento automático de expressoes faciais gramaticais na lingua brasileira de sinais*. PhD thesis, Universidade de São Paulo, 2011.

J. Gałka, M. Mąsior, M. Zaborski, e K. Barczewska. Inertial motion sensing glove for sign language gesture acquisition and recognition. *IEEE Sensors Journal*, 16(16):6310–6316, 2016.

A. Gandhi. Data augmentation: How to use deep learning when you have limited data — part 2. https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/, 2018.

W. Gao, G. Fang, D. Zhao, e Y. Chen. A chinese sign language recognition system based on sofm/srn/hmm. *Pattern Recognition*, 37(12):2389–2402, 2004.

A. Gesser. *Língua?: Que língua é essa?: crenças e preconceitos em torno da língua de sinais e da realidade surda*. Parábola Editorial, São Paulo, 2009. ISBN 978-85-7934-001-7.

A. K. S. Goes. Marcadores prosódicos da libras: o papel das expressões corporias. Master's thesis, Universidade Federal de Alagoas, Alagoas, Brasil, 3 2019.

R. R. Guerra. Deep Learning for Accessibility: Detection and Segmentation of Regions of Interest for Sign Language Recognition Systems. Trabalho Final de Curso, UFMG, 2019.

R. R. Guerra, T. M. Rezende, F. G. Guimaraes, e S. G. M. Almeida. Facial expression analysis in brazilian sign language for sign recognition. In *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*, pages 216–227. SBC, 2018.

D. Guo, W. Zhou, A. Li, H. Li, e M. Wang. Hierarchical recurrent deep fusion using adaptive clip summarization for sign language translation. *IEEE Transactions on Image Processing*, 29:1575–1590, 2019.

Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, e M. S. Lew. Deep learning for visual understanding: A review. *Neurocomputing*, 187:27–48, 2016.

Y. Guo, Y. Liu, T. Georgiou, e M. S. Lew. A review of semantic segmentation using deep neural networks. *International journal of multimedia information retrieval*, 7(2):87–93, 2018.

R. Gupta e A. Kumar. Indian sign language recognition using wearable sensors and multi-label classification. *Computers & Electrical Engineering*, 90:106898, 2021.

S. Hadfield e R. Bowden. Scene particles: Unregularized particle-based scene flow estimation. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):564–576, 2013.

M. M. Hasan e P. K. Misra. Brightness factor matching for gesture recognition system using scaled normalization. *International Journal of Computer Science & Information Technology*, 3(2), 2011.

M. Hassan, K. Assaleh, e T. Shanableh. Multiple proposals for continuous arabic sign language recognition. *Sensing and Imaging*, 20(1):4, 2019.

G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, e R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

B. Hisham e A. Hamouda. Arabic sign language recognition using ada-boosting based on a leap motion controller. *International Journal of Information Technology*, 13(3): 1221–1234, 2021.

E.-J. Holden, G. Lee, e R. Owens. Australian sign language recognition. *Machine Vision and Applications*, 16(5):312, 2005.

M. Honora e M. L. E. Frizanco. *Livro Ilustrado de Língua Brasileira de Sinais: Desvendando a Comunicação Usada Pelas Pessoas com Surdez*. Ciranda Cultural, São Paulo, 2010. ISBN 978-85-380-1421-8.

B. K. Horn e B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3): 185–203, 1981.

J. Huang, W. Zhou, H. Li, e W. Li. Sign language recognition using 3d convolutional neural networks. In *2015 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2015.

J. Huang, W. Zhou, H. Li, e W. Li. Attention-based 3d-cnns for large-vocabulary sign language recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9):2822–2832, 2018a.

J. Huang, W. Zhou, Q. Zhang, H. Li, e W. Li. Video-based sign language recognition without temporal segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018b.

S. Huang e Z. Ye. Boundary-adaptive encoder with attention method for chinese sign language recognition. *IEEE Access*, 9:70948–70960, 2021a. doi: 10.1109/ACCESS.2021.3078638.

S. Huang e Z. Ye. Boundary-adaptive encoder with attention method for chinese sign language recognition. *IEEE Access*, 9:70948–70960, 2021b.

S. Huang, C. Mao, J. Tao, e Z. Ye. A novel chinese sign language recognition method based on keyframe-centered clips. *IEEE Signal Processing Letters*, 25(3):442–446, 2018c.

A. Ibarguren, I. Maurtua, e B. Sierra. Layered architecture for real time sign recognition: Hand gesture and movement. *Engineering Applications of Artificial Intelligence*, 23(7): 1216–1228, 2010.

N. B. Ibrahim, M. M. Selim, e H. H. Zayed. An automatic arabic sign language recognition system (arslrs). *Journal of King Saud University-Computer and Information Sciences*, 30(4):470–477, 2018.

J. Imran e B. Raman. Deep motion templates and extreme learning machine for sign language recognition. *The Visual Computer*, 36(6):1233–1246, 2020.

I. Infantino, R. Rizzo, e S. Gaglio. A framework for sign language sentence recognition by commonsense context. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(5):1034–1039, 2007.

K. Inoue, T. Shiraishi, M. Yoshioka, e H. Yanagimoto. Depth sensor based automatic hand region extraction by using time-series curve and its application to japanese finger-spelled sign language recognition. *Procedia Computer Science*, 60:371–380, 2015.

Instituto Prominas. Libras - língua brasileira de sinais. Material Didático, 2017.

Intel Corporation. Opencv - open source computer vision library (versão 4.1.0), 2019. URL https://opencv.org/.

S. Islam, S. S. S. Mousumi, A. S. A. Rabby, S. A. Hossain, e S. Abujar. A potent model to recognize bangla sign language digits using convolutional neural network. *Procedia computer science*, 143:611–618, 2018.

S. Jadooki, D. Mohamad, T. Saba, A. S. Almazyad, e A. Rehman. Fused features mining for depth-based hand gesture recognition to classify blind human communication. *Neural Computing and Applications*, 28(11):3285–3294, 2017.

V. Jain, A. Jain, A. Chauhan, S. S. Kotla, e A. Gautam. American sign language recognition using support vector machine and convolutional neural network. *International Journal of Information Technology*, 13(3):1193–1200, 2021.

S. Ji, W. Xu, M. Yang, e K. Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1): 221–231, 2012.

X. Jiang, M. Lu, e S.-H. Wang. An eight-layer convolutional neural network with stochastic pooling, batch normalization and dropout for fingerspelling recognition of chinese sign language. *Multimedia Tools and Applications*, 79(21):15697–15715, 2020.

J. Jimenez, A. Martin, V. Uc, e A. Espinosa. Mexican sign language alphanumerical gestures recognition using 3d haar-like features. *IEEE Latin America Transactions*, 15 (10):2000–2005, 2017.

G. Joshi, S. Singh, e R. Vig. Taguchi-topsis based hog parameter selection for complex background sign language recognition. *Journal of Visual Communication and Image Representation*, 71:102834, 2020.

J. Joy, K. Balakrishnan, e M. Sreeraj. Signquiz: A quiz based tool for learning fingerspelled signs in indian sign language using aslr. *IEEE Access*, 2019.

P. R. M. Júnior, R. M. de Souza, R. d. O. Werneck, B. V. Stein, D. V. Pazinato, W. R. de Almeida, O. A. Penatti, R. d. S. Torres, e A. Rocha. Nearest neighbors distance ratio open-set classifier. *Machine Learning*, 106(3):359–386, 2017.

M. W. Kadous. Australian Sign Language signs (High Quality) Data Set, 2002. URL http://archive.ics.uci.edu/ml/datasets/Australian+Sign+Language+signs+(High+Quality).

M. W. Kadous e C. Sammut. Classification of multivariate time series and structured data using constructive induction. *Machine learning*, 58(2):179–216, 2005.

N. M. Kakoty e M. D. Sharma. Recognition of sign language alphabets and numbers based on hand kinematics using a data glove. *Procedia Computer Science*, 133:55–62, 2018.

Z. Katılmış e C. Karakuzu. Elm based two-handed dynamic turkish sign language (tsl) word recognition. *Expert Systems with Applications*, page 115213, 2021.

D. Kelly, J. Mc Donald, e C. Markham. Weakly supervised training of a sign language recognition system using multiple instance learning density matrices. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(2):526–541, 2010.

S. A. Khomami e S. Shamekhi. Persian sign language recognition using imu and surface emg sensors. *Measurement*, 168:108471, 2021.

T. Kim, J. Keane, W. Wang, H. Tang, J. Riggle, G. Shakhnarovich, D. Brentari, e K. Livescu. Lexicon-free fingerspelling recognition from video: Data, models, and signer adaptation. *Computer Speech & Language*, 46:209–232, 2017.

P. Kishore, D. A. Kumar, A. C. S. Sastry, e E. K. Kumar. Motionlets matching with adaptive kernels for 3-d indian sign language recognition. *IEEE Sensors Journal*, 18(8): 3327–3337, 2018.

E. S. Klima e U. Bellugi. Wit and poetry in american sign language. *Sign Language Studies*, 8(1):203–223, 1975.

O. Koller, J. Forster, e H. Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125, 2015.

O. Koller, S. Zargaran, H. Ney, e R. Bowden. Deep sign: Enabling robust statistical continuous sign language recognition via hybrid cnn-hmms. *International Journal of Computer Vision*, 126(12):1311–1325, 2018.

O. Koller, N. C. Camgoz, H. Ney, e R. Bowden. Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos. *IEEE transactions on pattern analysis and machine intelligence*, 42(9):2306–2320, 2019.

W. Kong e S. Ranganath. Signing exact english (see): Modeling and recognition. *Pattern Recognition*, 41(5):1638–1652, 2008.

W. Kong e S. Ranganath. Sign language phoneme transcription with rule-based hand trajectory segmentation. *Journal of Signal Processing Systems*, 59(2):211–222, 2010.

W. Kong e S. Ranganath. Towards subject independent continuous sign language recognition: A segment and merge approach. *Pattern Recognition*, 47(3):1294–1308, 2014.

V. E. Kosmidou e L. I. Hadjileontiadis. Using sample entropy for automated sign language recognition on semg and accelerometer data. *Medical & biological engineering & computing*, 48(3):255–267, 2010.

V. E. Kosmidou e L. J. Hadjileontiadis. Sign language recognition using intrinsic-mode sample entropy on semg and accelerometer data. *IEEE transactions on biomedical engineering*, 56(12):2879–2890, 2009.

V. E. Kosmidou, P. C. Petrantonakis, e L. J. Hadjileontiadis. Enhanced sign language recognition using weighted intrinsic-mode entropy and signer's level of deafness. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(6):1531–1543, 2011.

A. Krizhevsky, I. Sutskever, e G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012a.

A. Krizhevsky, I. Sutskever, e G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012b.

D. A. Kumar, A. Sastry, P. Kishore, e E. K. Kumar. 3d sign language recognition using spatio temporal graph kernels. *Journal of King Saud University-Computer and Information Sciences*, 2018a.

D. A. Kumar, A. Sastry, P. Kishore, e E. K. Kumar. Indian sign language recognition using graph matching on 3d motion captured signs. *Multimedia Tools and Applications*, 77(24):32063–32091, 2018b.

D. A. Kumar, A. Sastry, P. Kishore, E. K. Kumar, e M. T. K. Kumar. S3drgf: Spatial 3-d relational geometric features for 3-d sign language representation and recognition. *IEEE Signal Processing Letters*, 26(1):169–173, 2018c.

E. K. Kumar, P. Kishore, M. T. K. Kumar, D. A. Kumar, e A. Sastry. Three-dimensional sign language recognition with angular velocity maps and connived feature resnet. *IEEE Signal Processing Letters*, 25(12):1860–1864, 2018d.

E. K. Kumar, P. Kishore, A. Sastry, M. T. K. Kumar, e D. A. Kumar. Training cnns for 3-d sign language recognition with color texture coded joint angular displacement maps. *IEEE Signal Processing Letters*, 25(5):645–649, 2018e.

E. K. Kumar, P. Kishore, M. T. K. Kumar, e D. A. Kumar. 3d sign language recognition with joint distance and angular coded color topographical descriptor on a 2–stream cnn. *Neurocomputing*, 372:40–54, 2020.

P. Kumar, P. P. Roy, e D. P. Dogra. Independent bayesian classifier combination based sign language recognition using facial expression. *Information Sciences*, 428:30–48, 2018f.

S. Kumar, M. K. Bhuyan, e B. K. Chakraborty. Extraction of texture and geometrical features from informative facial regions for sign language recognition. *Journal on Multimodal User Interfaces*, 11(2):227–239, 2017.

C. C. Kuo, F. Glover, e K. S. Dhir. Analyzing and modeling the maximum diversity problem by zero-one programing. *Decision Sciences*, 24(6):1171–1185, 1993.

G. Latif, N. Mohammad, J. Alghazo, R. AlKhalaf, e R. AlKhalaf. ArASL: Arabic alphabets sign language dataset. *Data in brief*, 23:103777, 2019.

A. Le Guennec, S. Malinowski, e R. Tavenard. Data augmentation for time series classification using convolutional neural networks. In *ECML/PKDD workshop on advanced analytics and learning on temporal data*, 2016.

Y. LeCun, Y. Bengio, e G. Hinton. Deep learning. *nature*, 521(7553):436, 2015.

B. G. Lee e S. M. Lee. Smart wearable hand device for sign language interpretation system with sensors fusion. *IEEE Sensors Journal*, 18(3):1224–1232, 2017.

C. K. Lee, K. K. Ng, C.-H. Chen, H. C. Lau, S. Chung, e T. Tsoi. American sign language recognition and training method with recurrent neural network. *Expert Systems with Applications*, 167:114403, 2021.

D. Li, C. Rodriguez, X. Yu, e H. Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1459–1469, 2020.

S.-Z. Li, B. Yu, W. Wu, S.-Z. Su, e R.-R. Ji. Feature learning based on sae–pca network for human gesture recognition in rgbd images. *Neurocomputing*, 151:565–573, 2015a.

T.-H. S. Li, M.-C. Kao, e P.-H. Kuo. Recognition system for home-service-related sign language using entropy-based k-means algorithm and abc-based hmm. *IEEE transactions on systems, man, and Cybernetics: systems*, 46(1):150–162, 2015b.

W. Li. Webpage of dr wanqing li, 2017. URL http://www.uow.edu.au/~wanqing/#MSRAction3DDatasets.

W. Li, Z. Zhang, e Z. Liu. Action recognition based on a bag of 3d points. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 9–14. IEEE, 2010.

Y. Li, X. Chen, X. Zhang, K. Wang, e Z. J. Wang. A sign-component-based framework for chinese sign language recognition using accelerometer and semg data. *IEEE transactions on biomedical engineering*, 59(10):2695–2704, 2012.

Y. Li, X. Wang, W. Liu, e B. Feng. Deep attention network for joint hand gesture localization and recognition using static rgb-d images. *Information Sciences*, 441:66–78, 2018.

Y. Liao, P. Xiong, W. Min, W. Min, e J. Lu. Dynamic sign language recognition based on video sequence with blstm-3d residual networks. *IEEE Access*, 7:38044–38054, 2019.

K. M. Lim, A. W. C. Tan, C. P. Lee, e S. C. Tan. Isolated sign language recognition using convolutional neural network hand modelling and hand energy image. *Multimedia Tools and Applications*, pages 1–28, 2019.

L. Lin, B. Xu, W. Wu, T. W. Richardson, e E. A. Bernal. Medical time series classification with hierarchical attention-based temporal convolutional networks: A case study of myotonic dystrophy diagnosis. In *CVPR Workshops*, pages 83–86, 2019.

J. Liu, B. Liu, S. Zhang, F. Yang, P. Yang, D. N. Metaxas, e C. Neidle. Non-manual grammatical marker recognition based on multi-scale, spatio-temporal analysis of head pose and facial expressions. *Image and Vision Computing*, 32(10):671–681, 2014.

L. Liu e L. Shao. Learning discriminative representations from rgb-d video data. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

P. d. O. Lucas, M. A. Alves, P. C. de Lima e Silva, e F. G. Guimarães. Reference evapotranspiration time series forecasting with ensemble of convolutional neural networks. *Computers and Electronics in Agriculture*, 177:105700, 10 2020. ISSN 01681699. doi: 10.1016/j.compag.2020.105700.

Machine Vision Lab. IITR Sign Language Thermal Dataset 2018 (ISLTD2018) . https://www.iitr.ac.in/mvlab/documents/ISLTD2018_Download_Form.pdf, 2018.

P. F. Marentette. Its in her hands. a case study of the emergence of phonology in american sign language. Master's thesis, McGill University, Departament of Psychology, Montreal, 1995.

MCC Lab. SLR Dataset. http://mccipc.ustc.edu.cn/mediawiki/index.php/SLR_Dataset, 8 2020.

Microsoft. Kinect for Windows SDK 2.0: HighDetailFacePoints Enumeration. https://docs.microsoft.com/en-us/previous-versions/windows/kinect/dn791778(v=ieb.10), 2014.

Minds. Brazilian sign language recognition. http://minds.eng.ufmg.br/project/4, 2019.

M. Mohandes, M. Deriche, U. Johar, e S. Ilyas. A signer-independent arabic sign language recognition system using face detection, geometric features, and a hidden markov model. *Computers & Electrical Engineering*, 38(2):422–433, 2012.

M. A. Mohandes. Recognition of two-handed arabic signs using the cyberglove. *Arabian Journal for Science and Engineering*, 38(3):669–677, 2013.

P. Molchanov, S. Gupta, K. Kim, e J. Kautz. Hand gesture recognition with 3d convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1–7, 2015.

P. Nakjai e T. Katanyukul. Hand sign recognition for thai finger spelling: an application of convolution neural network. *Journal of Signal Processing Systems*, 91(2):131–146, 2019.

S. Nayak, S. Sarkar, e B. Loeding. Distribution-based dimensionality reduction applied to articulated motion recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(5):795–810, 2008.

T. D. Nguyen e S. Ranganath. Facial expressions in american sign language: Tracking and recognition. *Pattern Recognition*, 45(5):1877–1891, 2012.

M. Oszust e M. Wysocki. Point clouds corresponding to dynamic gestures registered by kinect, 2016a. URL http://vision.kia.prz.edu.pl/dynamickinect.php.

M. Oszust e M. Wysocki. Point clouds corresponding to dynamic gestures registered by time-of-flight (tof) camera, 2016b. URL http://vision.kia.prz.edu.pl/dynamictof.php.

C. Oz e M. C. Leu. American sign language word recognition with a sensory glove using artificial neural networks. *Engineering Applications of Artificial Intelligence*, 24(7): 1204–1213, 2011.

I. Pacifici, P. Sernani, N. Falcionelli, S. Tomassini, e A. F. Dragoni. A surface electromyography and inertial measurement unit dataset for the italian sign language alphabet. *Data in Brief*, 33:106455, 2020. ISSN 2352-3409. doi: https://doi.org/10.1016/j.dib.2020.106455. URL https://www.sciencedirect.com/science/article/pii/S2352340920313378.

J. Pan, Y. Luo, Y. Li, C.-K. Tham, C.-H. Heng, e A. V.-Y. Thean. A wireless multi-channel capacitive sensor system for efficient glove-based gesture recognition with ai at the edge. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 67(9):1624–1628, 2020a.

W. Pan, X. Zhang, e Z. Ye. Attention-based sign language recognition network utilizing keyframe sampling and skeletal features. *IEEE Access*, 8:215592–215602, 2020b.

P. Papapetrou, G. Kollios, S. Sclaroff, e D. Gunopulos. Mining frequent arrangements of temporal intervals. *Knowledge and Information Systems*, 21(2):133, 2009.

I. Papastratis, K. Dimitropoulos, D. Konstantinidis, e P. Daras. Continuous sign language recognition through cross-modal alignment of video and text embeddings in a joint-latent space. *IEEE Access*, 8:91170–91180, 2020.

A. F. K. Passos, W. P. Quintino, e L. F. da Silva. Constituição de sinais dos modos e pontos de articulação da língua portuguesa, em libras/constitution of signs of the modes and points of articulation of the portuguese language, in pounds. *Revista ECOS*, 24(1), 2018.

A. Patrícia Rocha, H. Miguel Pereira Choupina, M. d. C. Vilas-Boas, J. Maria Fernandes, e J. Paulo Silva Cunha. Body joints tracked by the kinect v2., Aug 2018. URL https://plos.figshare.com/articles/figure/Body_joints_tracked_by_the_Kinect_v2_/6932288/1.

H. Pedrini e W. R. Schwartz. *Análise de imagens digitais: princípios, algoritmos e aplicações*. Thomson Learning, 2008.

L. Pigou, S. Dieleman, P.-J. Kindermans, e B. Schrauwen. Sign language recognition using convolutional neural networks. In *European Conference on Computer Vision*, pages 572–578. Springer, 2014.

G. Pradhan, B. Prabhakaran, e C. Li. Hand-gesture computing for the hearing and speech impaired. *IEEE MultiMedia*, 15(2):20–27, 2008.

S. Qi, X. Wu, W.-H. Chen, J. Liu, J. Zhang, e J. Wang. semg-based recognition of composite motion with convolutional neural network. *Sensors and Actuators A: Physical*, 311: 112046, 2020.

R. M. Quadros e L. B. Karnopp. *Língua de Sinais Brasileira: Estudos Lingüísticos*. Artmed, Porto Alegre, 2004. ISBN 85-363-0308-5.

L. Quesada, G. López, e L. Guerrero. Automatic recognition of the american sign language fingerspelling alphabet to assist people living with speech or hearing impairments. *Journal of Ambient Intelligence and Humanized Computing*, 8(4):625–635, 2017.

T. Raghuveera, R. Deepthi, R. Mangalashri, e R. Akshaya. A depth-based indian sign language recognition using microsoft kinect. *Sādhanā*, 45(1):1–13, 2020.

G. A. Rao e P. Kishore. Selfie video based continuous indian sign language recognition system. *Ain Shams Engineering Journal*, 9(4):1929–1939, 2018.

R. Rastgoo, K. Kiani, e S. Escalera. Hand sign language recognition using multi-view hand skeleton. *Expert Systems with Applications*, page 113336, 2020.

S. Ravi, M. Suman, P. Kishore, K. Kumar, A. Kumar, et al. Multi modal spatio temporal co-trained cnns with single modal testing on rgb–d based sign language gesture recognition. *Journal of Computer Languages*, 52:88–102, 2019.

T. M. Rezende. Aplicação de técnicas de inteligência computacional para análise da expressão facial em reconhecimento de sinais de libras. Master's thesis, Universidade Federal de Minas Gerais, Programa de Pós Graduação em Engenharia Elétrica, Belo Horizonte, Minas Gerais, Brasil, 12 2016.

T. M. Rezende. Estudo de redes neurais convolucionais para o reconhecimento de sinais de libras. *ReseachGate*, 2019. doi: 10.13140/RG.2.2.20729.60000. URL http://rgdoi.net/10.13140/RG.2.2.20729.60000.

T. M. Rezende. Repositório GitLab. GitLab, 12 2020. URL https://gitlab.com/tamiresrezende.

T. M. Rezende, C. L. de Castro, e S. G. M. Almeida. An approach for brazilian sign language (bsl) recognition based on facial expression and k-nn classifier. In *29th SIBGRAPI, Workshop on Face Processing Applications, on Proceedings*, pages 1–2, 2016a.

T. M. Rezende, C. L. de Castro, F. A. O. Mota, C. A. L. Nametala, R. S. Corrêa, e S. G. M. Almeida. Reconhecimento de expressões faciais em sinais da língua brasileira de sinais (libras) utilizando os classificadores k-nn e svm. *Anais do XII Simpósio de Mecânica Computacional*, 2016b.

T. M. Rezende, C. L. Castro, e S. G. M. Almeida. Análise da expressao facial em reconhecimento de sinais de libras. In *XIII Simpósio Brasileiro de Automação Inteligente*, volume 13, pages 465–470, 2017.

T. M. Rezende, S. G. M. Almeida, e F. G. Guimarães. Development and validation of a Brazilian sign language database for human gesture recognition. *Neural Computing and Applications*, 3 2021. ISSN 0941-0643. doi: 10.1007/s00521-021-05802-4. URL http://link.springer.com/10.1007/s00521-021-05802-4.

E. Rho, K. Chan, E. J. Varoy, e N. Giacaman. An experiential learning approach to learning manual communication through a virtual reality environment. *IEEE Transactions on Learning Technologies*, 13(3):477–490, 2020.

C. H. Rodrigues. Situações de incompreensão vivenciadas por professor ouvinte e alunos surdos na sala de aula: processos interpretativos e oportunidades de aprendizagem. Master's thesis, Universidade Federal de Minas Gerais, Programa de Pós Graduação em Engenharia Elétrica, Belo Horizonte, Minas Gerais, Brasil, 2008.

V. Rodrigues. Métricas de avaliação: acurácia, precisão, recall... quais as diferenças? https://medium.com/@vitorborbarodrigues/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-recall-quais-as-diferen%C3%A7as-c8f05e0a513c, 2019.

M.-C. Roh e S.-W. Lee. Human gesture recognition using a simplified dynamic bayesian network. *Multimedia Systems*, 21(6):557–568, 2015.

F. Ronchetti, F. Quiroga, C. Estrebou, L. Lanzarini, e A. Rosete. Lsa64: A dataset for argentinian sign language, 2016. URL http://facundoq.github.io/unlp/lsa64/index.html.

S. Ruffieux, D. Lalanne, E. Mugellini, e O. A. Khaled. A survey of datasets for human gesture recognition. In *International Conference on Human-Computer Interaction*, pages 337–348. Springer, 2014.

T. Sahana, S. Paul, S. Basu, e A. F. Mollah. Hand sign recognition from depth images with multi-scale density features for deaf mute persons. *Procedia Computer Science*, 167:2043–2050, 2020.

M. Serrão, G. d. A. e Aquino, M. Costa, e C. F. F. Costa Filho. Human activity recognition from accelerometer with convolutional and recurrent neural networks. *Polytechnica*, pages 1–11, 2021.

F. Shah, M. S. Shah, W. Akram, A. Manzoor, R. Orban, e D. S. AbdElminaam. Sign language recognition using multiple kernel learning: A case study of pakistan sign language. *IEEE Access*, 2021.

A. Sharma, N. Sharma, Y. Saxena, A. Singh, e D. Sadhya. Benchmarking deep neural network approaches for indian sign language recognition. *Neural Computing and Applications*, pages 1–12, 2020.

S. M. Shohieb, H. K. Elminir, e A. Riad. Signsworld atlas; a benchmark arabic sign language database. *Journal of King Saud University-Computer and Information Sciences*, 27(1): 68–76, 2015.

G. F. Simons e C. D. Fennig. Ethnologue: Languages of the world, 2018. URL https://www.ethnologue.com/subgroups/sign-language.

K. Simonyan e A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014a.

K. Simonyan e A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014b.

O. M. Sincan e H. Y. Keles. Autsl: A large scale multi-modal turkish sign language dataset and baseline methods. *IEEE Access*, 8:181340–181355, 2020.

S. Skansi. *Introduction to Deep Learning: from logical calculus to artificial intelligence*. Springer, 2018.

T. Starner, J. Weaver, e A. Pentland. A wearable computer-based american sign language recogniser. *Personal Technologies*, 1(4):241–250, Dec 1997. doi: 10.1007/BF01682027. URL https://doi.org/10.1007/BF01682027.

T. Starner, J. Weaver, e A. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on pattern analysis and machine intelligence*, 20(12):1371–1375, 1998.

W. C. Stokoe. *Sign Language Structure: An Outline of the Visual Communication Systems of American Deaf*. University of Buffalo Press, New York, 1960.

C. Sun, T. Zhang, B.-K. Bao, C. Xu, e T. Mei. Discriminative exemplar coding for sign language recognition with kinect. *IEEE Transactions on Cybernetics*, 43(5):1418–1428, 2013.

M. Suneetha, M. Prasad, e P. Kishore. Multi-view motion modelled deep attention networks (m2da-net) for video based sign language recognition. *Journal of Visual Communication and Image Representation*, 78:103161, 2021.

S. Tamura e S. Kawasaki. Recognition of sign language motion images. *Pattern Recognition*, 21(4):343 – 353, 1988. ISSN 0031-3203. doi: https://doi.org/10.1016/0031-3203(88)90048-9. URL http://www.sciencedirect.com/science/article/pii/0031320388900489.

V. K. Tanwar, H. Buckchash, B. Raman, e R. Bhargava. Dense motion analysis of german finger spellings. *Multimedia Tools and Applications*, 78(8):9511–9536, 2019.

J. R. Terven e D. M. Córdova-Esparza. Kin2. a kinect 2 toolbox for matlab. *Science of Computer Programming*, 130:97–106, 2016.

N. Thiracitta, H. Gunawan, et al. Sibi sign language recognition using convolutional neural network combined with transfer learning and non-trainable parameters. *Procedia Computer Science*, 179:72–80, 2021.

A. S. Tolba e A. Abu-Rezq. Arabic glove-talk (agt): A communication aid for vocally impaired. *Pattern Analysis and Applications*, 1(4):218–230, 1998.

D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, e M. Paluri. C3D: generic features for video analysis. *CoRR*, abs/1412.0767, 2014. URL http://arxiv.org/abs/1412.0767.

P. Trigueiros, F. Ribeiro, e L. P. Reis. Generic system for human-computer gesture interaction: applications on sign language recognition and robotic soccer refereeing. *Journal of Intelligent & Robotic Systems*, 80(3-4):573–594, 2015.

N. Tubaiz, T. Shanableh, e K. Assaleh. Glove-based continuous arabic sign language recognition in user-dependent mode. *IEEE Transactions on Human-Machine Systems*, 45(4):526–533, 2015.

I. Y. Tyukin, A. N. Gorban, S. Green, e D. Prokhorov. Fast construction of correcting ensembles for legacy artificial intelligence systems: Algorithms and a case study. *Information Sciences*, 485:230–247, 2019.

H. Van der Hulst. Units in the analysis of signs. *Phonology*, 10(2):209–241, 1993.

T. Viéville e S. Crahay. Using an hebbian learning rule for multi-class svm classifiers. *Journal of Computational Neuroscience*, 17(3):271–287, 2004.

C. Vogler e S. Goldenstein. Facial movement analysis in asl. *Universal Access in the Information Society*, 6(4):363–374, 2008.

A. Wadhawan e P. Kumar. Deep learning-based sign language recognition system for static signs. *Neural Computing and Applications*, pages 1–12, 2020.

H. Wang, X. Chai, X. Hong, G. Zhao, e X. Chen. Isolated sign language recognition with grassmann covariance matrices. *ACM Transactions on Accessible Computing (TACCESS)*, 8(4):1–21, 2016.

H. Wang, X. Chai, e X. Chen. A Novel Sign Language Recognition Framework Using Hierarchical Grassmann Covariance Matrix. *IEEE Transactions on Multimedia*, 21(11): 2806–2814, 2019.

J. Wu, L. Sun, e R. Jafari. A wearable system for recognizing american sign language in real-time using imu and surface emg sensors. *IEEE journal of biomedical and health informatics*, 20(5):1281–1290, 2016.

L. Xia, C.-C. Chen, e J. K. Aggarwal. Human detection using depth information by kinect. In *CVPR 2011 workshops*, pages 15–22. IEEE, 2011.

Q. Xiao, Y. Zhao, e W. Huan. Multi-sensor data fusion for sign language recognition based on dynamic bayesian network and convolutional neural network. *Multimedia Tools and Applications*, pages 1–18, 2018.

Q. Xiao, M. Qin, P. Guo, e Y. Zhao. Multimodal fusion based on lstm and a couple conditional hidden markov model for chinese sign language recognition. *IEEE Access*, 7: 112258–112268, 2019.

Q. Xiao, X. Chang, X. Zhang, e X. Liu. Multi-information spatial–temporal lstm fusion continuous sign language neural machine translation. *IEEE Access*, 8:216718–216728, 2020a.

Q. Xiao, M. Qin, e Y. Yin. Skeleton-based chinese sign language recognition and generation for bidirectional communication between deaf and hearing people. *Neural networks*, 125: 41–55, 2020b.

B. Xu, S. Huang, e Z. Ye. Application of tensor train decomposition in s2vt model for sign language recognition. *IEEE Access*, 9:35646–35653, 2021.

H.-D. Yang e S.-W. Lee. Simultaneous spotting of signs and fingerspellings based on hierarchical conditional random fields and boostmap embeddings. *Pattern Recognition*, 43(8):2858–2870, 2010.

H.-D. Yang, S. Sclaroff, e S.-W. Lee. Sign language spotting with a threshold model based on conditional random fields. *IEEE transactions on pattern analysis and machine intelligence*, 31(7):1264–1277, 2008.

X. Yang, X. Chen, X. Cao, S. Wei, e X. Zhang. Chinese sign language recognition based on an optimized tree-structure framework. *IEEE journal of biomedical and health informatics*, 21(4):994–1004, 2016.

J.-W. Yoon, S.-I. Yang, e S.-B. Cho. Adaptive mixture-of-experts models for data glove interface with multiple users. *Expert Systems with Applications*, 39(5):4898–4907, 2012.

M. Zahedi, D. Keysers, T. Deselaers, e H. Ney. RWTH-BOSTON-50 Database, 2005. URL https://www-i6.informatik.rwth-aachen.de/web/Software/Databases/Signlanguage/details/rwth-boston-50/index.php.

S. Zhang, W. Meng, H. Li, e X. Cui. Multimodal spatiotemporal networks for sign language recognition. *IEEE Access*, 7:180270–180280, 2019.

X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang, e J. Yang. A framework for hand gesture recognition based on accelerometer and emg sensors. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(6):1064–1076, 2011.

R. Zhao e A. M. Martinez. Labeled graph kernel for behavior analysis. *IEEE transactions on pattern analysis and machine intelligence*, 38(8):1640–1650, 2015.

T. Zhao, J. Liu, Y. Wang, H. Liu, e Y. Chen. Towards low-cost sign language gesture recognition leveraging wearables. *IEEE Transactions on Mobile Computing*, 2019.

Y. Zhou, X. Chen, D. Zhao, H. Yao, e W. Gao. Adaptive sign language recognition with exemplar extraction and map/ivfs. *IEEE signal processing letters*, 17(3):297–300, 2009.

Apêndice A

Parâmetros Fonológicos da Língua de Sinais

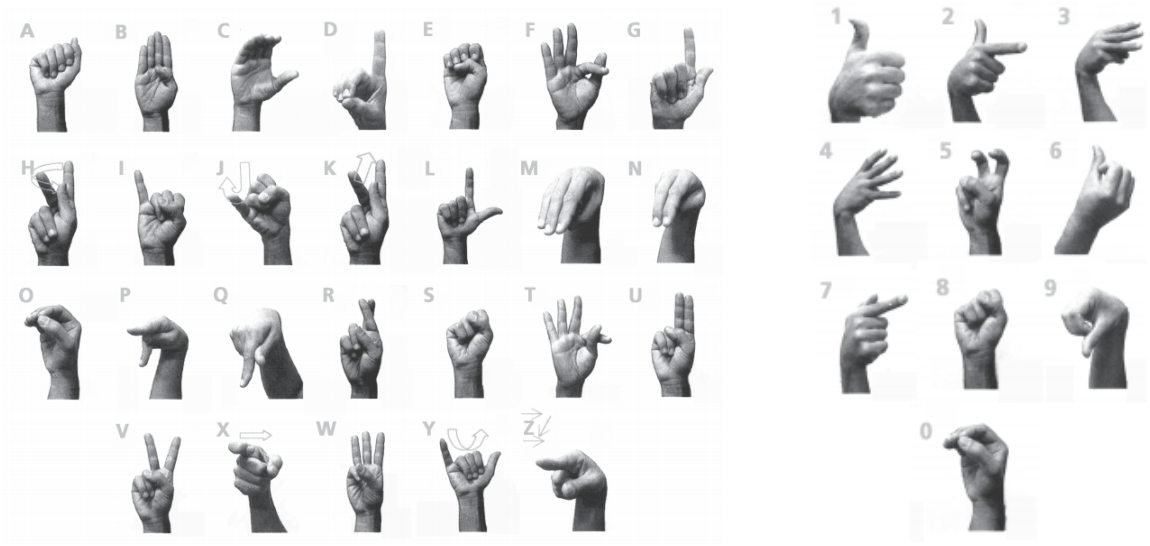
O reconhecimento linguístico da língua de sinais teve início com Willian Stokoe em 1960 (Stokoe, 1960). Ele foi o primeiro linguista a pesquisar as partes constituintes dos sinais da Língua Americana de Sinais (ASL), com foco em sua organização fonológica. Inicialmente ele propôs que um sinal na ASL fosse composto por 3 parâmetros: configuração de mão (CM), ponto de articulação ou locação (PA) e movimento da mão (M). Isto significa que estes seriam os fonemas que constituem a língua de sinais, de forma análoga aos fonemas das línguas orais (Quadros e Karnopp, 2004).

Após o trabalho de Stokoe, verificou-se que a orientação da palma da mão (Or) e as expressões não-manuais (ENM) também auxiliavam na formação de um sinal (Battison, 1974, 1978). A partir daí estes dois parâmetros fonológicos foram adicionados à língua de sinais. Apesar destes estudos serem realizados na língua americana de sinais, Van der Hulst (1993) mostrou que esta estrutura fonológica é universal e pode ser compartilhada em todas as línguas de sinais.

A.1 Configuração de Mão

A configuração de mão refere-se à forma que as mãos assumem durante a execução do sinal. Há sinais em que ambas as mãos produzem o movimento em conjunto e outros que utilizam apenas uma delas, também denominada de mão dominante. Ela pode ser tanto a mão direita no caso do sinalizador ser destro, quanto a esquerda, quando ele é canhoto. A CM pode ser representada por uma letra do alfabeto (Figura 70a), por números (Figura 70b) ou por formas manuais diversas. Vale ressaltar também que a utilização deste parâmetro abrange a datilologia[1] de palavras que não possuem sinal específico na língua e nomes próprios.

[1] Datilologia é similar a soletração na língua oral. Representação de cada letra.



(a) Letras (b) Números

Figura 70 – Alfabeto manual da Libras.

Adaptado de: Felipe (2009).

De acordo com Ferreira-Brito (1995), a Libras possui 46 CM's. Entretanto, estudos mais recentes apresentam 64 configurações (Felipe, 2009), como mostra a Figura 71. Em Passos et al. (2018) é realizada uma análise deste parâmetro e fica explícito que ainda não há um consenso da comunidade científica em relação à quantidade de CM's.

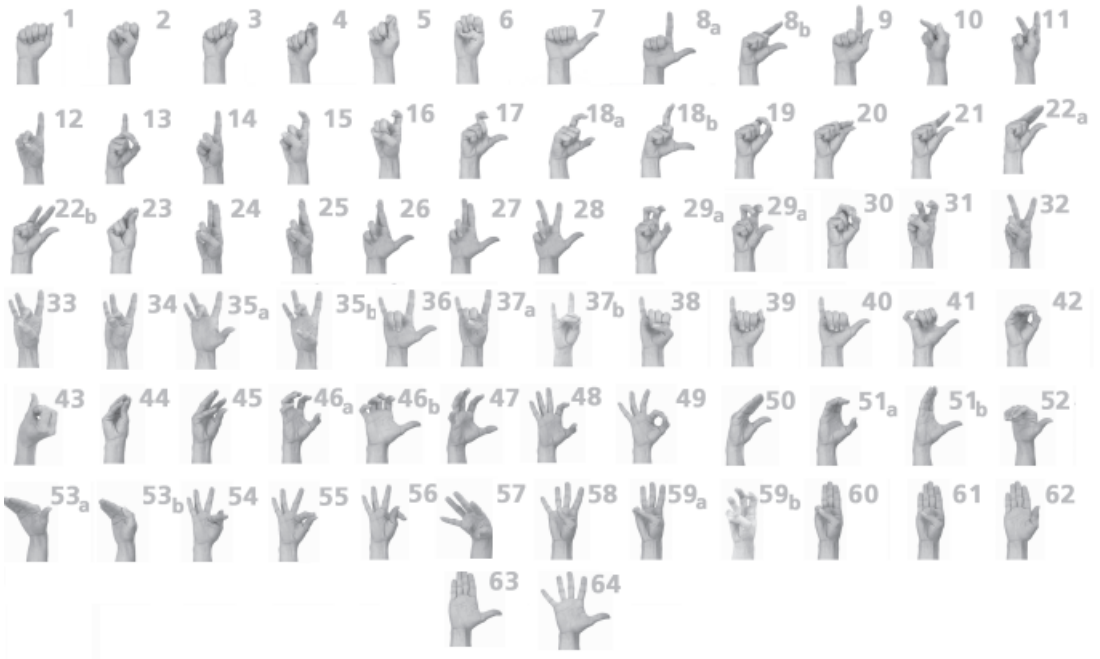


Figura 71 – Configurações de mão da Libras.

Adaptado de: Felipe (2009).

A.2 Ponto de Articulação ou Locação

Ponto de articulação ou locação é a área do corpo ou do espaço em que se realiza o sinal e as principais áreas de articulação dos sinais compreendem a região de alcance das mãos. Vale ressaltar que, como as línguas de sinais são tridimensionais, é permitida a utilização do espaço à frente do sinalizador para a construção dos sinais e frases. A Figura 72 apresenta exemplos de PA. No sinal "baleia" (Figura 72a) o ponto de articulação é tocando o topo da cabeça e no sinal "escova de dente" (Figura 72b) é à frente da boca. Os principais PA's se dividem em cabeça, tronco, braço, mão e espaço neutro:

- Cabeça: topo da cabeça, testa, rosto, parte superior do rosto, parte inferior do rosto, orelha, olhos, nariz, boca, bochechas, queixo;
- Tronco: pescoço, ombro, busto, estômago, cintura;
- Braço: braço, antebraço, cotovelo, pulso;
- Mão: palma e costas da mão, lado do indicador, lado do dedo mínimo, dedos (anular, médio, indicador, polegar, mínimo), pontas dos dedos; e
- Espaço neutro: espaço livre localizado à frente do corpo.

(a) PA: topo da cabeça. (b) PA: boca.

Figura 72 – Exemplos de ponto de articulação em Libras: sinal (a) "baleia" e (b) "escova de dente".

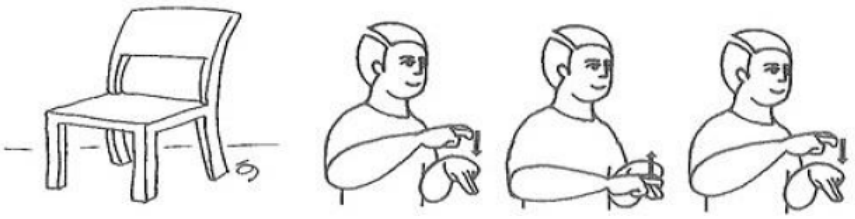
Fonte: Honora e Frizanco (2010).

A.3 Movimento

Esse parâmetro é referente aos movimentos da mão, do pulso e os movimentos direcionais no espaço (Klima e Bellugi, 1975). Ele serve para distinguir itens lexicais[2]

[2] Léxico: conjunto de palavras existente em um determinado idioma, que as pessoas têm à disposição para expressar-se, oralmente ou por escrito em seu contexto.

e pode estar relacionado com a direcionalidade dos verbos e suas variações em relação ao tempo. A Figura 73 mostra o movimento realizado nos sinais "cadeira" e "sentar". A configuração da mão é idêntica em ambos os sinais, mas eles realizam movimentos distintos. No sinal "cadeira", o movimento é curto e com duas repetições e no "sentar-se" o movimento é longo e sem repetição.



(a) M: retilíneo, bidirecional e repetido.

(b) M: retilíneo, unidirecional e longo.

Figura 73 – Exemplos de movimento em Libras: sinal (a) "cadeira" e (b) "sentar-se".

Adaptado de: Capovilla et al. (2012a,b).

Os movimentos identificados na Libras referem-se ao tipo, direcionalidade, maneira e frequência. Assim, Fereira-Brito (1990) categorizou cada um destes atributos com detalhes:

- Tipo:
  - contorno ou forma geométrica: retilíneo, helicoidal, circular, semicircular, sinuoso, angular, pontual;
  - interação: alternado, de aproximação, de separação, de inserção, cruzado;
  - contato: de ligação, de agarrar, de deslizamento, de toque, de esfregar, de riscar, de escovar ou de pincelar;
  - torcedura do pulso: rotação, com refreamento;
  - dobramento do pulso: para cima, para baixo;
  - situação da parte interna das mãos: abertura, fechamento, curvamento e dobramento simultâneos/gradativos;
- Direcionalidade:
  - unidirecional: para cima, para baixo, para a direita, para a esquerda, para dentro, para fora, para o centro, para a lateral inferior esquerda, para lateral

inferior direita, para a lateral superior esquerda, para lateral superior direita, para específico ponto referencial;

- bidirecional: para cima e para baixo, para a esquerda e para a direita, para dentro e para fora, para laterais opostas - superior direita e inferior esquerda;
- não-direcional;

- Maneira:
  - categoria que descreve a qualidade, tensão e velocidade do movimento: contínuo, de retenção ou refreado; e
- Frequência de repetição: simples, repetido.

A.4 Orientação da Mão

Orientação da mão é a direção para a qual a palma da mão aponta durante a execução do sinal, sejam elas: para cima, para baixo, para dentro (para o corpo), para fora (para frente), para direita (para o lado ipsilateral) e para esquerda (para o lado contralateral), como ilustra a Figura 74. Este foi um dos parâmetros adicionados a língua de sinais após o estudo de Stokoe, ao perceberem que ele era capaz de distinguir sinais como os ilustrados na Figura 75, "ajudar alguém" e "ser ajudado".

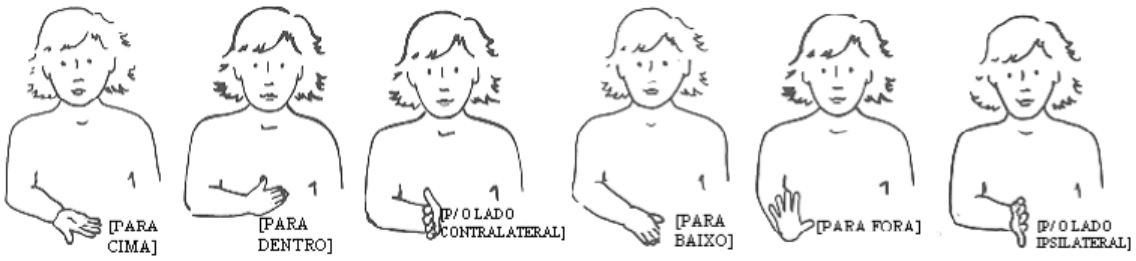


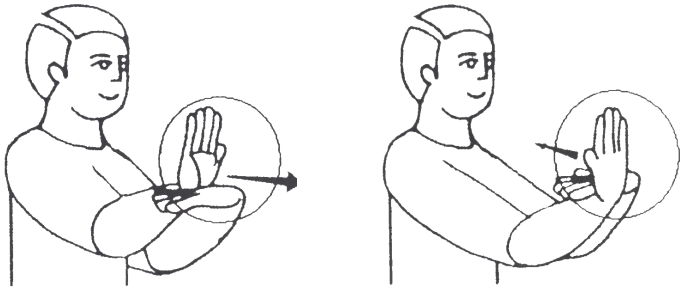
Figura 74 – Orientação da palma da mão em Libras.

Adaptado de: Marentette (1995).

A.5 Expressões Não-Manuais

As expressões não-manuais referem-se ao movimento da face, da cabeça ou do tronco. São elementos gramaticais que desempenham diferentes papéis na língua como, por exemplo, marcação de sentenças interrogativas relacionadas a "sim-não" (Figura 76) e sentenças interrogativas relativas às palavras ou iniciadas com "qu" (que, quem, quando).

De acordo com Ferreira-Brito (1995) os atributos relativos às expressões não-manuais podem ocorrer simultaneamente e são categorizados da seguinte forma:



(a) Or: para fora. (b) Or: para dentro.

Figura 75 – Exemplos de orientação da mão em Libras: sinal (a) "ajudar alguém" e (b) "ser ajudado".

Fonte: Capovilla e Raphael (2004).

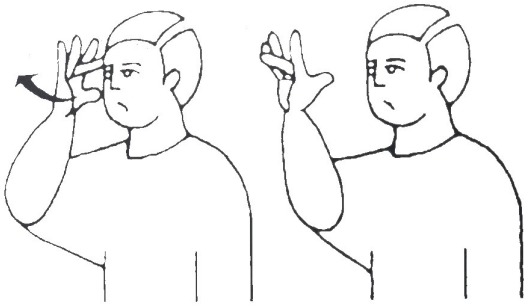


Figura 76 – Expressão facial negativa do sinal "não sei".

Fonte: Capovilla e Raphael (2004).

- Cabeça: balanceamento para frente e para trás (sim), balanceamento para os lados (não), inclinação para frente, inclinação para o lado, inclinação para trás;
- Rosto:
  - parte superior: sobrancelhas franzidas, olhos arregalados, lance de olhos, sobrancelhas levantadas;
  - parte inferior: bochechas infladas, bochechas contraídas, lábios contraídos e projetados e sobrancelhas franzidas, correr da língua contra a parte inferior interna a bochecha, apenas bochecha direita inflada, contração do lábio superior, franzir do nariz;
- Rosto e cabeça: cabeça projetada para frente, olhos levemente cerrados, sobrancelhas franzidas; cabeça projetada para trás e olhos arregalados; e
- Tronco: para frente, para trás, balanceamento alternado dos ombros, balanceamento simultâneo dos ombros, balanceamento de um único ombro.

Apêndice B

Descrição dos Sinais Gravados

Nesse apêndice encontram-se as definições de cada sinal que compõe a base de dados utilizada nesse trabalho e a descrição de cada um dos 5 parâmetros que caracterizam o sinal (Tabela 17). Todas estas informações tiveram como base os trabalhos de Capovilla et al. (2012a,b, 2017a,b,c).

1. Acontecer: Suceder, ocorrer, realizar-se;
2. Aluno: Indivíduo que recebe instrução ou educação em estabelecimentos de ensino ou não; discípulo, estudante, escolar;
3. Amarelo: Da cor do sol, da cor da gema de ovo, da cor do ouro, da cor do losango da bandeira brasileira;
4. América: Continente geográfico dividido em Norte, Central e Sul. Está localizado entre dois oceanos (Atlântico e Pacífico) e se estende de norte a sul do planeta;
5. Aproveitar: Tirar proveito de alguma oportunidade ou pessoa sem prejudicá-la (sentido positivo);
6. Bala: Pequena porção de açúcar refinado em ponto vítreo e que contém, usualmente, substâncias aromáticas e corantes;
7. Banco: Estabelecimento particular ou estatal cujas atividades consistem na guarda, empréstimo ou investimento de dinheiro, transações com títulos de créditos, entre outros;
8. Banheiro: Aposento sanitário equipado com vaso sanitário, lavatório e boxe de banho com chuveiro ou banheira;
9. Barulho: Som irregular, desarmonioso, desagradável e muito intenso, gerador de desconforto no indivíduo que o ouve;

10. Cinco: Cardinal designativo de cindo unidades, ou seja, meia dezena;

11. Conhecer: Saber. Estar informado. Ter conhecimento de. Ter informação de. Ter experiência com. Estar familiarizado com. Ter ideia ou noção de. Ser perito ou versado em. Ser capaz de reconhecer e discernir (algo);

12. Espelho: Superfície polida que reflete, com grande quantidade, a luz ou a imagem dos objetos. Lâmina de vidro ou cristal, estanhada na parte superior para adorno ou para as pessoas se verem;

13. Esquina: Canto onde duas vias públicas se cortam ou cruzam;

14. Filho: Descendente masculino ou feminino de um pai e uma mãe, em relação a eles. Descendente natural, em primeiro grau, de uma pessoa em relação a ela;

15. Maça: Fruto da macieira, pertencente a família das rosáceas, com mais de seiscentas variedades;

16. Medo: Sentimento de inquietação resultante da ideia de um perigo real ou da presença de alguma coisa estranha ou perigosa. Apreensão. Receio. Temor;

17. Ruim: Mau, moral ou fisicamente. Destituído de mérito. O que não possui serventia. Que perdeu valor, que não tem valor, que não presta. Corrupto, podre, estragado. Imoral. Inferior. Nocivo. Malvado, perverso;

18. Sapo: Anfíbio sem cauda, que se desenvolve na água e que apresenta, na fase adulta, hábitos terrestres, procurando a água na época da reprodução;

19. Vacina: Preparado microbiano atenuado que, introduzido no organismo, produz reações imunológicas e formação de anticorpos que tomam o organismo imune aos micróbios usados;

20. Vontade: Anseio. Apetite. Aspiração. Cobiça. Desejo. Disposição de ânimo, urgente e apetitiva. Empenho determinado. Desejo que precisa ser satisfeito. Impulso deliberado. Interesse.

Tabela 17 – Descrição dos sinais gravados. Notação: Mão Direita (MD), Mão Esquerda (ME).

Sinal	CM	PA	M	Or	ENM
Acontecer	ME aberta, MD em "V"	À frente do tronco	Mover a MD em direção à esquerda, batendo o dorso da mão na palma esquerda	MD e ME para cima	–
Aluno	MD em "A"	Ombro esquerdo	Tocar a parte superior do braço	Para trás	–
Amarelo	MD em "D"	À frente da testa	Indicador tocando a testa e abaixando até o nariz	Para a esquerda	–
América	Mãos abertas, unidas pelos polegares	À frente	–	ME frente e MD para trás	–
Aproveitar	MD em "I"	À frente	Mover a mão para baixo, girando a palma para cima	Para baixo	–
Bala	MD em "1", indicador curvado	Ao lado da bochecha direita	Tocar a bochecha com o dedo indicador (2 vezes)	Para esquerda	Bochecha distendida pela língua
Banco	Mão aberta	Ao lado do pescoço	Bater com a ponta dos dedos no lado direito do pescoço (2 vezes)	Para Baixo	–
Banheiro	MD com dedos indicador e mínimo esticados; ME fechada	Dedos indicador e mínimo tocando o antebraço esquerdo	Aproximar e afastar	Para baixo (ambas)	–
Barulho	MD em "1", indicador curvado	Ao lado da orelha direita	Mover a MD para baixo e para a direita	Para baixo	Expressão facial contraída

Continue na próxima página

Tabela 17 – *Continue na página anterior*

Cinco	MD em "V", dedos indicador e médio curvados	À frente do corpo	–	Para frente	–
Conhecer	MD em "4"	À frente do queixo	Bater a lateral do indicador próximo ao lado direito do queixo	Para a esquerda	–
Espelho	MD aberta	À frente do rosto	Girar para esquerda e para direita	Para trás	–
Esquina	MD aberta com os dedos apontando para baixo, ME fechada (Braço esquerdo horizontal dobrado diante do peito)	À frente do tronco	Passar a palma dos dedos sobre o baço esquerdo, do ombro em direção ao pulso	MD para trás e ME para baixo	–
Filho	MD aberta, dedos separados	Lado esquerdo do peito	Mover a mão para frente, unindo as pontas dos dedos	Para trás	–
Maça	Mão em "C" horizontal	À frente da boca	Girar a palma para cima	Para trás	–
Medo	Mão horizontal aberta	Lado esquerdo do peito	Unha do dedo médio tocando a palma do polegar, dorso do polegar tocando o lado esquerdo do peito. Distender o dedo médio várias vezes	Para trás	Expressão de temor
Ruim	Mão aberta	Abaixo do queixo	Dorso da mão sob o queixo, oscilando os dedos	Para baixo	Franzindo a testa

Continue na próxima página

Tabela 17 – *Continue na página anterior*

Sapo	Mãos abertas (braço esquerdo horizontal dobrado na frente do corpo	À frente do tronco	Mover a MD em direção ao cotovelo esquerdo, tocando o braço esquerdo durante o movimento	Para baixo	Inflando várias vezes as bochechas
Vacina	MD fechada, dedos polegar, indicador e médio distendidos, curvados e apontando para trás	Parte superior do braço esquerdo	Mover o polegar, lentamente, em direção aos demais dedos	Para direita	–
Vontade	Mão em "1"	Abaixo do queixo	Passar a ponta do indicador para baixo sobre o pescoço (2 vezes)	Para trás	–

Fonte: Capovilla et al. (2012a,b, 2017a,b,c)

Apêndice C

Publicações

C.1 Facial Expression Analysis in Brazilian Sign Language for Sign Language.

XV Encontro Nacional de Inteligência Artificial e Computacional - ENIAC 2018.

22 a 25 de outubro de 2018.

São Paulo - São Paulo.

DOI: 10.5753/eniac.2018.4418

Facial Expression Analysis in Brazilian Sign Language for Sign Recognition

Rúbia Reis Guerra[1], **Tamires Martins Rezende**[2], **Frederico Gadelha Guimarães**[2], **Sílvia Grasiella Moreira Almeida**[3]

[1]Departamento de Engenharia Elétrica – Universidade Federal de Minas Gerais Belo Horizonte – Minas Gerais – Brasil

[2]Programa de Pós-Graduação em Engenharia Elétrica – Universidade Federal de Minas Gerais – Belo Horizonte – Minas Gerais – Brasil

[3]Instituto Federal de Minas Gerais – Campus Ouro Preto – Ouro Preto – Minas Gerais – Brasil

{rubia-rg,tamiresrezende,fredericoguimaraes}@ufmg.br
silvia.almeida@ifmg.edu.br

Abstract. *Sign language is one of the main forms of communication used by the deaf community. The language's smallest unit, a "sign", comprises a series of intricate manual and facial gestures. As opposed to speech recognition, sign language recognition (SLR) lags behind, presenting a multitude of open challenges because this language is visual-motor. This paper aims to explore two novel approaches in feature extraction of facial expressions in SLR, and to propose the use of Random Forest (RF) in Brazilian SLR as a scalable alternative to Support Vector Machines (SVM) and k-Nearest Neighbors (k-NN). Results show that RF's performance is at least comparable to SVM's and k-NN's, and validate non-manual parameter recognition as a consistent step towards SLR.*

1. Introduction

The first studies of sign language structure date back to 1960, with Stokoe [Landar and Stokoe 1961]. Sign language is a form of visual-motor communication used by the deaf community. Its smallest unit, a "signal", comprises non-manual parameters, movement of the face, eyes, head and torso, and manual parameters such as hand configuration, palm orientation, location and movement. Similar to spoken languages, sign languages have distinct grammatical structures, varying by country and culture [Gesser 2009]. According to [Laborit 1998], any concept can be expressed by means of signals without any loss of content.

Although the first mentions of sign language dissemination in Brazil date back to the 19th century [de Assis Silva 2012], just recently in Brazil it has been sanctioned as an official language. Brazilian Sign Language (LIBRAS) was only recognized in 2002 as the country's second official language, by law number 10.436 [Brasil 2002].

In the past decade alone, the improvement of machine learning techniques has led to significant advancements in automatic speech recognition. Speech-based Natural User Interfaces (NUI) were made possible and widely spread, facilitating human-human and human-machine interaction [López et al. 2017]. Despite recent progress, as seen in [Hinton et al. 2012] and [Pigou et al. 2015], sign language recognition still lags behind. In particular, Brazilian sign language recognition has only recently been explored ([Rezende et al. 2016], [Freitas et al. 2017], [Filho et al. 2017]).

Most of the past work in LIBRAS recognition have focused primarily on the manual parameters of signals ([Almeida et al. 2014], [Dias et al. 2009], [Freitas et al. 2017]). This study is an extension of [Rezende et al. 2016], which proposes recognition of LIBRAS signs through facial expression. Facial components of signals are represented by the movement of the head, eye, eyebrow, etc., and are grammatical elements that make up the structure of the language, emphasizing and intensifying the signs when necessary.

There are over 10,000 signs in LIBRAS, and facial expressions are not mandatory in all signals [Capovilla 2017]. Moreover, different signals can share similar expressions. Hence, realizing signal classification tasks solely based on facial expression, in which each output label corresponds to a sign, does not generalize well for a large vocabulary. [Almeida et al. 2014] proposes a more scalable solution by extracting structural components of a signal, such as hand configuration and type of movement. Classification is then performed within each component, limiting the range of possible output classes. Signs can be recognized among others by grouping the individual results obtained for each component and comparing to a predetermined reference, such as [Capovilla 2017]. A similar approach is proposed in this work: instead of considering each signal's direct meaning as an output class, here we propose that each facial expression is labeled according to the closest fundamental expression (neutral, happy, sad, angry, fearful, surprised, disgusted) or according to the most prominent feature (e.g. tongue out or sucked cheeks).

The dataset used in this work is the same as presented in [Rezende et al. 2016], and comprises a descriptor of facial and manual spatial coordinates, and summarized frames of each signals' recordings [Rezende et al. 2016]. In this work, only coordinates pertaining to facial points were considered. In addition to experiments utilizing facial coordinates, classification is performed on two novel descriptors:

- Points selected by inspection of video frames;
- Facial points which suffer most displacement throughout frames.

Past studies in LIBRAS recognition have employed Support Vector Machines (SVM), k-Nearest Neighbors (k-NN) and artificial neural networks for classification tasks ([Porfirio et al. 2013], [Filho et al. 2017], [Almeida et al. 2014], [Rao et al. 2017]). In this work, due to the high dimensionallity of the available data, Random Forest [Breiman 2001] was proposed as a classification method in the learning stage. The Random Forest algorithm implicitly performs feature selection [Breiman 2001] and have consistently shown robust results throughout a plethora of applications, including facial expression recognition [Pu et al. 2015]. Furthermore, the algorithm yields good accuracy results in classification tasks when compared to other state-of-the-art methods [Zhang et al. 2017], is fast to train and can easily be parallelized [Genuer et al. 2017], posing as a scalable candidate for learning tasks on a larger collection of LIBRAS signs.

2. Related Work

Most literature on signal language recognition deal with the relative configuration of the hands [Escobedo-Cardenas and Camara-Chavez 2015],

[Almeida et al. 2014], [de Paula Neto et al. 2015], [Pariwat and Seresangtakul 2017], [Uddin and Chowdhury 2016]. Most of these studies carry out recognition of the alphabet in their respective languages. Few works focus just on the information of the non-manual expressions [Freitas et al. 2017], [Rezende et al. 2016], [Uddin 2015], seeking to emphasize the importance of facial expressions in sentences and even the recognition of signs. Although they achieve high rates of recognition of expressions, the common methodologies proposed are adequate only to the set of data used. This therefore limits replicability, because there is no way to universalize the dataset in the literature. In addition, recognizing a sign using only one of its parameters is unfeasible, since multiple signs may share the same configuration of a parameter, while others do not use it at all.

One of the greatest challenges of SLR is dealing with all parameters simultaneously, since each sign language has its own unique grammatical structure and some signs may incorporate only a few parameters. Performing recognition using more than one parameter is the work of [Rao et al. 2017] and [Yang and Lee 2011].

In [Rao et al. 2017], a real-time signaling system was implemented using the frontal camera of a cell phone. Twenty signs, among words and letters, were tested. The videos were recorded at a rate of 30fps and each signal was subjected to pre-processing, segmentation and feature extraction techniques. At the end of the process, the signals were labeled using an Artificial Neural Network approach. The developed application returned an audio from the performed signal. Despite the well-structured methodology, the sample size was small. Another weakness of the system is the number of neurons in the hidden layer. The value was chosen by trial and error, preventing recognition of new, unseen samples, since the parameters of the system would have to be recalculated.

In the work of [Yang and Lee 2013], the manual segments of signals were identified and then analyzed in regard to their configurations. Facial expression was investigated if there were any ambiguities related to the analysis of the hands. The database used has 24 signals, of which 17 are related to the alphabet and 5, to facial expressions, making up 98 sentences from the American Sign Language. Recognition is therefore accomplished in separate steps for each parameter, hand or face.

As seen, works presented in the literature do not have a common methodological mechanism that is capable of evaluating the real situations that a user of sign language encounters. Overcoming this barrier enables the creation of a system for instantaneous recognition of signals, facilitating communication with others who do not know the language. Thus, the intention of this paper is to propose a generalizable approach to SLR, by focusing on classifying each signal's basic parameters, instead of attempting to extract its direct meaning.

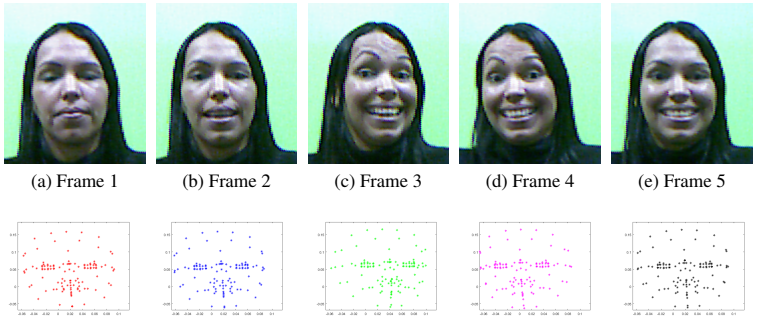
2.1. Facial Expression Recognition

Automated analysis of human emotions is a multidisciplinary endeavor and a key component of human-computer interaction. A multitude of approaches have been studied in an effort to capture the nuances that differentiate facial expressions [Zeng et al. 2009].

The problem is well-studied and bench-marked within Computer Vision, presenting a variety of consolidated databases [Gross 2005]. Research directions differ, among a plethora of factors, in the choice of data, sentiment categorization, temporal or static analysis and learning method ([Du et al. 2014], [Yu and Zhang 2015], [Jung et al. 2015], [Abdullah et al. 2014]).

2.2. LIBRAS Data

The data utilized in this work were obtained from [Rezende et al. 2016] and contain recordings of 10 different signs, represented by non-manual parameters of the Brazilian Sign Language (LIBRAS). Figure 1 shows five frames of a recording of the "happiness" sign. Each sign (to calm down, to accuse, to annihilate, to love, to gain weight, happiness, slim, lucky, surprise and angry) has 10 recordings, totaling a database of 100 samples.



(a) Frame 1 (b) Frame 2 (c) Frame 3 (d) Frame 4 (e) Frame 5

(f) Frame 1-Points (g) Frame 2-Points (h) Frame 3-Points (i) Frame 4-Points (j) Frame 5-Points

Figure 1. Frames from fourth recording of the sign "happiness".

The signs were captured using a RGB-D sensor (Microsoft Kinect) and processed by nuiCaptureAnalyze software. In the processing stage, each recording's images (Figures 1a to 1e) and xy-coordinates of 121 points located across the face were obtained (Figures 1f to 1j). This work focuses on the 121 points, which served as base descriptors for each facial expression.

Each sign in the original dataset was mapped to one of the labels as presented in Table 1, taking into consideration the closest fundamental expression or the most prominent facial parameter [Capovilla 2017]:

Table 1. Sign mapping

Sign	New label	Sign	New label
To calm down	Neutral	Happiness	Happy
To accuse	Angry	Skinny	Sucked cheeks
To annihilate	Angry	Lucky	Neutral
In love	Happy	Surprised	Surprised
To fatten	Inflated cheeks	Angry	Angry

3. Feature Extraction

In [Rezende et al. 2016], classification was performed on four different feature vectors composed by the 121 facial points:

- Utilizing raw data;
- Performing z-normalization on (x,y) coordinates separately, for each recording of each signal;
- Normalizing each xy-coordinate of each signal's recordings in relation to the centroid of the first corresponding frame;
- Normalizing each xy-coordinate of each signal's recordings in relation to the centroid of its current frame.

Due to data being highly dimensional in all aforementioned experiments (1210 features $\times$ 100 samples), this work proposes two novel approaches aiming to reduce feature space, discussed in the next subsections.

3.1. Points Selected by Inspection

Pairs of points were selected through inspection of video frames, tentatively capturing prominent characteristics of facial expressions. Each pair and its respective description is shown on Table 2 and the selected points can be seen in Figure 2.

Table 2. Selected pairs

Points	Description
6 and 3	Nose tip to mid supraorbital ridge
6 and 11	Nose tip to mid chin
8 and 9	Mouth opening height
20 and 25	Right eye opening
49 and 16	Outer eyebrows corners (left and right)
50 and 17	Eyebrow upper midpoint (left and right)
53 and 58	Left eye opening
65 and 32	Mouth opening width
91 and 92	Nasolabial folds (left and right)

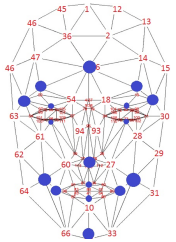


Figure 2. Selected points

Data dimension was reduced to 170 features (xy-coordinates of 17 points, 5 frames) $\times$ 100 samples. The resulting feature vector is as follows:

$$\underbrace{x_{i,j}}_{\text{signal i, recording j}} = \left[\underbrace{(s_x, s_y)_{1,1}\ (s_x, s_y)_{2,1}\ \ldots\ (s_x, s_y)_{s,f}}_{\text{point s coordinates, frame f}} \right] \tag{1}$$

3.2. Displacement Ranking

The original data set was processed following the steps bellow:

1. For each recording, a cumulative measure of displacement based on the Euclidean distance of a point in each consecutive frame was calculated;
2. Points were ranked within each signal, following the highest displacement value;
3. The first decile of each recording's rank was sampled, yielding 12 points with highest displacement per recording;
4. For each signal, a mode the first decile was determined;
5. Each signal's recordings were re-sampled from the original data with respect to each signal's mode. The new data set's dimensions are 120 features (xy-coordinates of 12 points, 5 frames) $\times$ 100 observations.

The resulting feature vector is as follows:

$$\underbrace{x_{i,j}}_{\text{signal i, recording j}} = \left[\underbrace{(p_x, p_y)_{1,1}\ (p_x, p_y)_{2,1}\ \ldots\ (p_x, p_y)_{p,f}}_{\text{point p coordinates, frame f}} \right] \tag{2}$$

4. Experimental Analysis

Three experiments were formulated to evaluate the proposed approaches:

1. Classification utilizing all 121 points;
2. Classification on the feature vector consisting of points selected by inspection of the video frames;
3. Classification on the reduced data set created through the displacement ranking.

All experiments were performed utilizing three classifiers, for means of comparison: Random Forest [Breiman 2001], Support Vector Machines [Cortes and Vapnik 1995] and k-Nearest Neighbors [Patrick and Fischer 1970]. The general structure of the solution is as seen on Algorithm 1. For each classifier, 30 models were constructed in each experiment, resulting in 90 models per experiment. Further performance measures were derived from predicted values and discussed later in this section.

Algorithm 1: Sign Classification

input : sign samples
output: predicted expression

```
maxIt i ← 30;
for i ← 1 to maxIt do
    randomization of samples;
    train i ← 80% of each class;
    test i ← 20% of each class;
    parameters ← k-fold cross-validation;
    model ← classifier(train i, parameters);
    predictions i ← model(test i);
end
```

4.1. Random Forest

Breiman's Random Forest (RF, [Breiman 2001]) is a powerful ensemble approach based on bootstrap aggregation of multiple decision trees, and is widely utilized in applications in which the number of features exceeds the number of observations. The algorithm differentiates from other decision tree methods in the way that at each node

split in the learning process, the feature space is randomly sampled with replacement. A final prediction is obtained by aggregating all constructed trees through majority voting [Boulesteix et al. 2012].

The Random Forest algorithm has several tuning parameters, of which most show high dependency on the data set [Ließ et al. 2012]. Hence, one of the practical challenges when using RF is parameterization. In this work, Random Forests parameters are selected through randomized search stratified 3-fold cross validation, obtaining 500 different settings [Pedregosa et al. 2011]. The search space was limited to according to the values shown in Table 3.

Table 3. Random Forest parameters

Parameter description	Possible values
Number of trees	[800, 2000], step size = 10
Number of features (p) considered at each split	$\{\log_2 p, \sqrt{p}, 0.3p\}$
Maximum depth of a tree	{[10, 80], step size = 10; or None}
Minimum number of samples required to split a node	{3, 5, 7};
Minimum number of samples required at each leaf-node	{2, 3, 4}
Whether bootstrapping occurs when constructing trees	{'True', 'False'}
Split quality measure	{'Gini', 'Entropy'}

4.2. Support Vector Machine

The SVM (Support Vector Machine) classifier presented by [Cortes and Vapnik 1995] is currently considered the state-of-art in classification and regression problems [Zhang et al. 2017]. The SVM algorithm finds points that make up support vectors, which, in turn, compose a hyperplane that optimizes the distance between the classes, serving as a decision boundary. This boundary is obtained using training data, and is applied to classify the test data.

The SVM solves binary classification problems. However, LIBRAS recognition is multiclass problem. Package e1071's implementation of SVM [Meyer and Wien 2001] uses a one-against-one technique for multiclass problems, thus, it was selected to be utilized in this work. In addition, package e1071 has tools to perform automatic search, by cross-validation, of the cost parameters C and gamma(γ), relative to the separation surface of the classes. [Hsu et al. 2016] advises that the parameter C to vary from 2^{-5} to 2^{15}, and γ, from 2^{-15} to 2^{3}. In relation to the kernel, Radial Base Function (RBF) was used, chosen according to [Hsu et al. 2016], after taking into consideration the cases tested in this work.

4.3. k-NN

The use of k-NN [Patrick and Fischer 1970] for classification problems is well established in the literature. Due to the classifier's efficiency in terms of running time [Zhang et al. 2017], it was included in this work for comparison with SVM's and RF's results. To determine the class of a sample *m* not belonging to the training set, the k-NN classifier looks for *k* elements of the training set that are closest to *m*, and assigns its class based on which class represents the majority of these selected *k* elements.

With the selected training data, cross-validation was used to find the value for *k* that provided the highest accuracy rate (k_{best}). Thus, the training data were divided into 5-folds of the same size and 5 cross-validation iterations were performed applying the leave-one-out technique.

4.4. Results and Discussion

After applying the procedure shown in Algorithm 1, accuracy results were obtained for each classifier. Results were statistically analyzed utilizing ANOVA, Shapiro-Wilk and Fligner-Killeen tests [Elliott and Woodward 2007], and can be seen in Figure 3.

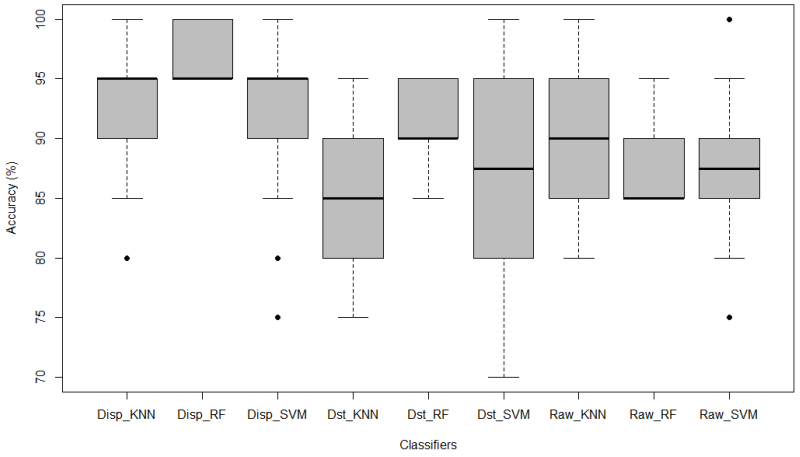


Figure 3. Comparison of results for each classifier

For classification with raw data, k-NN obtained the best average accuracy (89.33%). However, classifiers' overall performance are comparable. Utilizing points selected through inspection, Random Forest obtained the best average accuracy (91.33%) and achieved best overall performance within the experiment. SVM and k-NN presented similar performances. Finally, in the experiment with displacement ranking data, Random Forest presented the highest average accuracy (96.67%) and performed similarly to SVM, both surpassing k-NN's results.

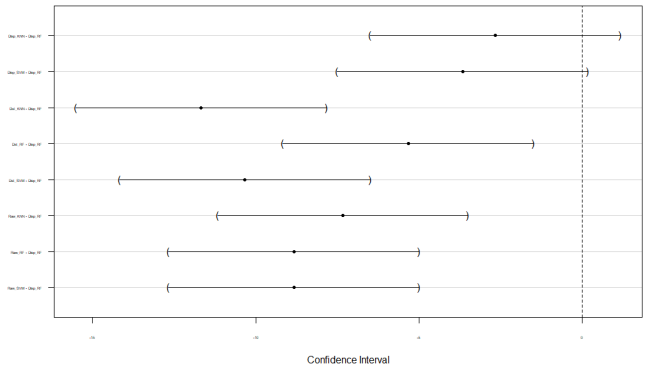


Figure 4. One-Against-All confidence interval of Random Forest + Displacement Ranking

Through an all-against-all analysis, as shown in Figure 4, the combination of displacement ranking and either Random Forest or SVM classifiers has shown the best overall results. Class-wise results for Random Forest and displacement ranking, seen in Table 4, show that the expression "Inflated Cheeks" had the lowest accuracy results, and was misclassified as "Happy".

Table 4. Aggregated confusion matrix for Random Forest + Displacement Ranking

Predicted	Angry	Happy	Surprised	Neutral	Sucked Cheeks	Inflated Cheeks	All
Angry	180	0	0	0	0	0	180
Happy	0	100	0	0	0	20	120
Surprised	0	0	60	0	0	0	60
Neutral	0	0	0	120	0	0	120
Sucked Cheeks	0	0	0	0	60	0	60
Inflated Cheeks	0	0	0	0	0	60	60
All	180	100	60	120	60	80	600

The work presented in this paper shows the following advancements when compared to [Rezende et al. 2016]: (i) validates Random Forest as an scalable alternative to SVM and k-NN; (ii) corroborates a new, generalizable approach to LIBRAS recognition, that can be combined to [Almeida et al. 2014] constituting a hollistic method to SLR.

5. Conclusion

This study proposed a new approach to LIBRAS recognition. In contrast to works presented in the literature, facial parameters were analyzed and classification was performed identifying basic elements that make up the structure of the language. Results validate the approach and introduce Random Forest as a good candidate for learning tasks. Since non-manual configurations may be shared by different signs - or may not be used at all - future works addressing both manual and non-manual parameters are expected to deliver a holistic, and more precise, solution to SLR.

It is of fundamental importance that Computational Intelligence minimizes communication barriers and facilitates communication between those who have hearing impairments with those who do not. LIBRAS is not a compulsory component of the basic school curriculum at present, therefore sign language literacy level is low, making it hard for the deaf to communicate with the majority of the population.

6. Acknowledgments

This work has been supported by the Brazilian agency CAPES.

References

Abdullah, M. F. A., Sayeed, M. S., Muthu, K. S., Bashier, H. K., Azman, A., and Ibrahim, S. Z. (2014). Face recognition with symmetric local graph structure (SLGS). *Expert Systems with Applications*, 41(14):6131–6137.

Almeida, S. G. M., Guimarães, F. G., and Ramírez, J. A. (2014). Feature extraction in brazilian sign language recognition based on phonological structure and using RGB-d sensors. *Expert Systems with Applications*, 41(16):7259–7271.

Boulesteix, A.-L., Janitza, S., Kruppa, J., and König, I. R. (2012). Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(6):493–507.

Brasil (2002). Lei n^o 10.436, de 24 de abril de 2002.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Capovilla, F. C. (2017). *Dicionário da Língua de Sinais do Brasil. A Libras em Suas Mãos - 3 Volumes*. Edusp.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

de Assis Silva, C. A. (2012). Igreja católica e surdez: território, associação e representação política. *Religião & Sociedade*, 32(1):13–38.

de Paula Neto, F. M., Cambuim, L. F., Macieira, R. M., Ludermir, T. B., Zanchettin, C., and Barros, E. N. (2015). Extreme learning machine for real time recognition of brazilian sign language. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE.

Dias, D. B., Madeo, R. C. B., Rocha, T., Biscaro, H. H., and Peres, S. M. (2009). Hand movement recognition for brazilian sign language: A study using distance-based neural networks. In *2009 International Joint Conference on Neural Networks*. IEEE.

Du, S., Tao, Y., and Martinez, A. M. (2014). Compound facial expressions of emotion. *Proceedings of the National Academy of Sciences*, 111(15):E1454–E1462.

Elliott, A. and Woodward, W. (2007). *Statistical Analysis Quick Reference Guidebook*. SAGE Publications, Inc.

Escobedo-Cardenas, E. and Camara-Chavez, G. (2015). A robust gesture recognition using hand local data and skeleton trajectory. In *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE.

Filho, C. F. F. C., de Souza, R. S., dos Santos, J. R., dos Santos, B. L., and Costa, M. G. F. (2017). A fully automatic method for recognizing hand configurations of brazilian sign language. *Research on Biomedical Engineering*, 33(1):78–89.

Freitas, F. A., Peres, S. M., Lima, C. A. M., and Barbosa, F. V. (2017). Grammatical facial expression recognition in sign language discourse: a study at the syntax level. *Information Systems Frontiers*, 19(6):1243–1259.

Genuer, R., Poggi, J.-M., Tuleau-Malot, C., and Villa-Vialaneix, N. (2017). Random forests for big data. *Big Data Research*, 9:28–46.

Gesser, A. (2009). *LIBRAS?: Que língua é essa?: crenças e preconceitos em torno da língua de sinais e da realidade surda*. Parábola Editorial, São Paulo.

Gross, R. (2005). Face databases. In S.Li, A., editor, *Handbook of Face Recognition*. Springer, New York.

Hinton, G., Deng, L., Yu, D., Dahl, G., rahman Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97.

Hsu, C., Chang, C., and Lin, C. (2016). A practical guide to support vector classification.

Jung, H., Lee, S., Yim, J., Park, S., and Kim, J. (2015). Joint fine-tuning in deep neural networks for facial expression recognition. In *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE.

Laborit, E. (1998). *The cry of the gull*. Gallaudet University Press, Washington, DC.

Landar, H. and Stokoe, W. C. (1961). Sign language structure: An outline of the visual communication systems of the american deaf. *Language*, 37(2):269.

Ließ, M., Glaser, B., and Huwe, B. (2012). Uncertainty in the spatial prediction of soil texture. *Geoderma*, 170:70–79.

López, G., Quesada, L., and Guerrero, L. A. (2017). Alexa vs. siri vs. cortana vs. google assistant: A comparison of speech-based natural user interfaces. In *Advances in Intelligent Systems and Computing*, pages 241–250. Springer International Publishing.

Meyer, D. and Wien, T. U. (2001). Support vector machines. the interface to libsvm in package e1071. online-documentation of the package e1071 for r.

Pariwat, T. and Seresangtakul, P. (2017). Thai finger-spelling sign language recognition using global and local features with SVM. In *2017 9th International Conference on Knowledge and Smart Technology (KST)*. IEEE.

Patrick, E. and Fischer, F. (1970). A generalized k-nearest neighbor rule. *Information and control*, 16(2):128 – 152.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pigou, L., Dieleman, S., Kindermans, P.-J., and Schrauwen, B. (2015). Sign language recognition using convolutional neural networks. In *Computer Vision - ECCV 2014 Workshops*, pages 572–578. Springer International Publishing.

Porfirio, A. J., Wiggers, K. L., Oliveira, L. E., and Weingaertner, D. (2013). LIBRAS sign language hand configuration recognition based on 3d meshes. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE.

Pu, X., Fan, K., Chen, X., Ji, L., and Zhou, Z. (2015). Facial expression recognition from image sequences using twofold random forest classifier. *Neurocomputing*, 168:1173–1180.

Rao, G. A., Kishore, P. V. V., Sastry, A. S. C. S., Kumar, D. A., and Kumar, E. K. (2017). Selfie continuous sign language recognition with neural network classifier. In *Proceedings of 2nd International Conference on Micro-Electronics, Electromagnetics and Telecommunications*, pages 31–40. Springer Singapore.

Rezende, T. M., de Castro, C. L., and Almeida, S. G. M. (2016). An approach for brazilian sign language (bsl) recognition based on facial expression and k-nn classifier. In Fábio A. M. Cappabianco, Fábio A. Faria, J. A. and Körting, T. S., editors, *Conference on Graphics, Patterns and Images (SIBGRAPI '16)*. Sociedade Brasileira de Computação.

Uddin, M. A. and Chowdhury, S. A. (2016). Hand sign language recognition for bangla alphabet using support vector machine. In *2016 International Conference on Innovations in Science, Engineering and Technology (ICISET)*. IEEE.

Uddin, M. T. (2015). An ada-random forests based grammatical facial expressions recognition approach. In *2015 International Conference on Informatics, Electronics & Vision (ICIEV)*. IEEE.

Yang, H.-D. and Lee, S.-W. (2011). Combination of manual and non-manual features for sign language recognition based on conditional random field and active appearance model. In *2011 International Conference on Machine Learning and Cybernetics*. IEEE.

Yang, H.-D. and Lee, S.-W. (2013). Robust sign language recognition by combining manual and non-manual features based on conditional random field and support vector machine. *Pattern Recognition Letters*, 34(16):2051–2056.

Yu, Z. and Zhang, C. (2015). Image based static facial expression recognition with multiple deep network learning. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction - ICMI '15*. ACM Press.

Zeng, Z., Pantic, M., Roisman, G., and Huang, T. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58.

Zhang, C., Liu, C., Zhang, X., and Almpanidis, G. (2017). An up-to-date comparison of state-of-the-art classification algorithms. *Expert Systems with Applications*, 82:128–150.

C.2 Desenvolvimento de uma Base de Dados de Sinais de Libras para Aprendizado de Máquina: Estudo de Caso com CNN 3D.

14º Simpósio Brasileiro de Automação Inteligente - SBAI 2019.

27 a 30 de outubro de 2019.

Ouro Preto - Minas Gerais.

DOI: 10.17648/sbai-2019-111451

Desenvolvimento de uma Base de Dados de Sinais de Libras para Aprendizado de Máquina: Estudo de Caso com CNN 3D ⋆

Giulia Z. de Castro * **Rúbia R. Guerra** ***
Moises M. de Assis *** **Tamires M. Rezende** *
Gabriela T. B. de Almeida *** **Sílvia G. M. Almeida** **
Cristiano L. de Castro *** **Frederico G. Guimarães** ***

* *Programa de Pós-Graduação em Engenharia Elétrica - Universidade Federal de Minas Gerais, Belo Horizonte, Minas Gerais, Brasil, (e-mail: giuliaz@ufmg.br, tamiresrezende@ufmg.br).*
** *Instituto Federal de Minas Gerais - Campus Ouro Preto, Ouro Preto, Minas Gerais, Brasil, (e-mail: silvia.almeida@ifmg.edu.br)*
*** *Machine Intelligence and Data Science (MINDS) Laboratory, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil, (e-mail: rubia-rg@ufmg.br, moisesmendes@ufmg.br, crislcastro@ufmg.br, fredericoguimaraes@ufmg.br)*

Abstract: A recurrent problem in Brazilian Sign Language (Libras) recognition is the absence of a robust dataset that allows the validation of different methodologies. This work presents a new dataset for Libras and its respective recording procedure. The first available version contains 20 signs, recorded 5 times by 10 different signers, making up 1000 recordings. A study case with the new data utilizing a 3D Convolutional Neural Network for sign recognition is also presented, employing summarization and data augmentation techniques. The network implemented achieved an average accuracy of 72,6%.

Resumo: Um dos problemas enfrentados nos trabalhos de reconhecimento de sinais de Libras é a ausência de uma base de dados robusta, que permita a validação de diferentes metodologias. Dessa forma, este trabalho apresenta uma nova base de dados da Língua Brasileira de Sinais e seu protocolo de gravação. A primeira versão da base contém 20 sinais, gravados 5 vezes, por cada um de 10 sinalizadores, totalizando 1000 gravações. É apresentado também um estudo de caso da aplicação de uma Rede Neural Convolucional 3D para a tarefa de classificação dos sinais, utilizando técnicas de sumarização e data augmentation. A rede implementada alcançou um resultado médio de 72,6% de acerto.

Keywords: Libras, sign recognition, dataset, recording protocol, 3DCNN.

Palavras-chaves: Libras, reconhecimento de sinais, base de dados, protocolo de gravação, CNN 3D.

1. INTRODUÇÃO

A língua de sinais é uma forma de comunicação visual-motora utilizada pela comunidade surda. Assim como as línguas orais, as línguas de sinais são únicas para cada cultura, apresentando estruturas gramaticais próprias. No Brasil, ela é chamada de Língua Brasileira de Sinais (Libras) e é a segunda língua oficial do país desde a publicação da Lei nº 10.436 em 2002.

O problema de identificação automática de sinais [1] das línguas de sinais pode ser considerado como uma aplicação específica do problema mais geral de reconhecimento de gestos, visto que as expressões não-manuais não são necessárias em grande parte dos sinais (Capovilla, 2017). Geralmente, o *framework* padrão de reconhecimento de gestos consiste na extração de atributos espaço-temporais de quadros de vídeo, seguida pela modelagem da dinâmica intra-quadros por meio de classificadores, como *Support Vector Machines* (SVM) (Rautaray e Agrawal, 2015), *Hidden Markov Model* (HMM) (Kumar et al., 2017) e Redes Neurais Artificiais (John et al., 2016).

Um dos grandes desafios em reconhecimento de Libras é a disponibilidade de bases de dados representativas que possibilitem a validação de novas metodologias. Até o momento, observa-se que a maioria dos trabalhos no assunto criaram suas próprias bases (Filho et al., 2017; Escobedo-Cardenas e Camara-Chavez, 2015; Almeida et al., 2014), o que dificulta a comparação das técnicas utilizadas na tarefa de reconhecimento. Uma das poucas bases disponíveis publicamente é a de Amaral et al. (2019), contendo informações de profundidade (Figura 1) para 10 sinais.

⋆ O presente trabalho foi realizado com o apoio financeiro da CAPES e CNPq.

[1] Menor unidade da língua de sinais, composta pelo movimento das mãos, corpo e expressões faciais.

Figura 1. Exemplo de imagem disponibilizada por Amaral et al. (2019)

Em relação ao registro de sinais de Libras, estudos anteriores utilizam sensores multimodais, como, por exemplo, o sensor RGB-D (Almeida et al., 2014; Escobedo-Cardenas e Camara-Chavez, 2015; Filho et al., 2017). Outras propostas abordam o problema por meio de vestimentas especiais, como luvas e sensores do tipo *wearable* (Kawamoto et al., 2018). Estes têm como objetivo contornar limitações causadas pela variação de iluminação, facilitar a segmentação das regiões de interesse para a detecção do sinal ou extrair atributos relativos à trajetória das mãos. O presente estudo busca contribuir com uma base de sinais padronizada, disponibilizada gratuitamente e em um cenário que permite adaptações por parte do usuário. Ela contém, inicialmente, 20 sinais gravados sistematicamente 5 vezes por 10 sinalizadores distintos, totalizando 1000 amostras e encontra-se disponível em Almeida et al. (2019).

Procurando estabelecer um patamar de comparação para trabalhos futuros aplicados a esta nova base de dados, decidiu-se aplicar um modelo de aprendizado baseado em redes neurais como um estudo de caso. Trabalhos propondo a utilização da CNN 2D em tarefas de classificação envolvendo imagens vêm ganhando atenção (Rawat e Wang, 2017), especialmente após os resultados obtidos por Krizhevsky et al. (2012) no dataset *ImageNet*. Abordagens de CNN 2D em vídeo, contudo, geralmente são aplicadas a cada quadro individual e não consideram a informação de movimento contida em múltiplos quadros contíguos. Uma forma de efetivamente incluir a correlação temporal inter-quadro no modelo é a utilização de convoluções 3D nas camadas convolucionais da CNN (Tran et al., 2014). Sendo assim, inspirado nas arquiteturas de CNN 3D propostas em Molchanov et al. (2015) e Zhang et al. (2017), este estudo apresenta uma abordagem para reconhecimento de sinais de Libras incorporando a implementação de uma CNN 3D ao *framework* padrão de reconhecimento de gestos.

O artigo está estruturado da seguinte forma: a seção 2 apresenta o protocolo de criação da base de sinais de Libras. Posteriormente a seção 3 descreve a arquitetura da CNN utilizada para classificação dos sinais. Por fim, os resultados são apresentados e discutidos nas seções 4 e 5, respectivamente.

2. BASE DE DADOS DE SINAIS DE LIBRAS

A construção ou a escolha de uma base de dados é uma etapa fundamental em qualquer problema de reconhecimento de padrões. Quando se trata de sinais de Libras, este assunto se torna desafiador, pois a maioria dos trabalhos criam as próprias bases de dados para validar suas metodologias. Realizando uma breve pesquisa bibliográfica, verificou-se alguns desses trabalhos. A Tabela 1 sintetiza as características de cada um.

Tabela 1. Bases de dados de língua de sinais.

Base	Ano	Reconhecimento de	Nº de amostras
Athitsos et al. (2008)	2008	Sinais da ASL*	3800
Li (2017)	2012	Gestos manuais de ASL	336
Conly et al. (2013)	2013	Sinais da ASL	1113
Almeida (2014)	2014	Sinais da Libras	170
Rezende (2016)	2016	Expressões Faciais da Libras	100

*Língua Americana de Sinais.

Tendo em vista que a Libras tem mais de 10 mil verbetes, percebe-se que há uma carência na área quando se trata de uma base com sinais de Libras padronizados e que sejam disponibilizados em um formato que permita a validação de sistemas de classificação computacional de forma robusta. Dessa forma, o presente trabalho desenvolveu um protocolo de gravação com base nos estudos de Ruffieux et al. (2014), Almeida (2014) e Rezende (2016). O protocolo aborda a escolha dos sinais e quem os executará, os sensores e *softwares* utilizados para aquisição dos vídeos, o cenário das gravações e a estrutura dos dados disponibilizados.

2.1 Seleção dos sinais

Com o auxílio de uma especialista em Libras foram selecionados 20 sinais da língua com base na diversidade das características dos parâmetros fonológicos da Libras. Estes parâmetros referem-se às unidades formacionais de um sinal e são caracterizados por: configuração da mão [2], ponto de articulação [3], movimento das mãos, orientação da palma da mão e expressões não-manuais. Partindo desses critérios, os sinais selecionados foram: acontecer, aluno, amarelo, América, aproveitar, bala, banco, banheiro, barulho, cinco, conhecer, espelho, esquina, filho, maçã, medo, ruim, sapo, vacina e vontade. Cada um deles foi gravado 5 vezes por cada um dos 10 sinalizadores [4].

2.2 Sinalizadores

Dentre os sinalizadores há homens e mulheres, surdos e ouvintes, sem distinção de vestimenta e raça, e com conhecimento variando de básico a avançado na Libras. Sugeriu-se aos sinalizadores que permanecessem olhando para a câmera em posição de descanso antes e após a execução do sinal, para marcar o início e o fim de cada gravação. Além disso, em todas as gravações a posição do sinalizador é fixa, em pé no centro do vídeo, e ele inicia e finaliza o sinal com as mãos sobre as pernas.

2.3 Cenário de gravação

No estúdio de gravação, o sensor ficou em posição fixa, gravando o movimento corporal superior, a expressão facial e o movimento manual. O arranjo do cenário foi disposto como apresentado na Figura 2. As gravações dos sinais ocorreram em um estúdio com boa iluminação e com plano

[2] Forma assumida pela mão na articulação do sinal.
[3] Área do corpo em que o sinal é articulado.
[4] Quem executa o sinal.

de fundo constante feito de tecido Chroma Key [5]. Este plano permite ao usuário da base remover ou adicionar diferentes fundos, sendo uma possível técnica para testar o desempenho de algoritmos de reconhecimento que usam padrões visuais.

Figura 2. Cenário de gravação.

2.4 Estrutura dos dados

Com o intuito de disponibilizar os vídeos dos sinais no formato MP4, utilizou-se uma câmera digital profissional [6], que possui uma taxa de gravação de 30 fps [7]. Cada quadro dos vídeos possui 1920 × 1080 *pixels*. A base de dados inicialmente proposta possui 1000 amostras [8]. Entretanto, houve o comprometimento de 5 gravações de um sinal [9], que foram descartadas, totalizando 995 amostras.

3. ARQUITETURA E PLANEJAMENTO EXPERIMENTAL

Neste trabalho foi realizado um pré-processamento dos dados e classificação por meio de uma CNN 3D. O pré-processamento consistiu em reduzir informações redundantes nos vídeos de cada gravação e aplicar uma ferramenta para aumentar o número de amostras. A primeira ação padroniza o número de imagens que representam cada gravação e a segunda fornece maior insumo de dados à CNN, que precisa de um grande volume para seu treinamento. A classificação dos sinais da base de dados é realizada pelos procedimentos esquematizados na Figura 3 .

3.1 Pré-processamento

Os vídeos da base de dados descrita foram submetidos a dois processos antes de serem passados à CNN 3D: sumarização e *data augmentation*.

A sumarização de vídeo foi aplicada neste trabalho com o objetivo de eliminar informações redundantes e uniformizar as gravações para terem o mesmo tamanho. No caso dos sinais da base utilizada, o número de quadros de cada sinal varia devido às diferentes velocidades de gravação dos sinalizadores, o que gera, dentre outros, quadros sequenciais muito semelhantes. Além disso, a sumarização escolhe um número pré-definido de quadros por vídeo, o que os torna padronizados e garante que os vetores de características da rede neural terão o mesmo tamanho para todas as amostras. A técnica de sumarização utilizada baseia-se no Problema da Diversidade Máxima (PDM) (Kuo et al., 1993), um problema de otimização que busca encontrar um conjunto de elementos para os quais a diversidade entre eles seja máxima. A solução adotada para o PDM foi a de Almeida et al. (2015), que resolve o problema por meio de uma estratégia evolutiva. Para este estudo escolheu-se utilizar uma sumarização de 12 quadros e, após sumarizar, foram excluídos o primeiro e o último quadros, pois verificou-se que neles a pessoa estava parada, isto é, caracterizam o início e o fim das gravações. Portanto, ao final da sumarização e do corte, obteve-se 10 quadros por vídeo. Este valor teve como referência os trabalhos de Rezende (2016) e Almeida (2014), utilizando o dobro de quadros pra obter mais detalhes da execução dos sinais.

Data augmentation refere-se a estratégias utilizadas para aumentar o volume de dados. Elas têm sido aplicadas no treinamento de CNNs por diversos autores para evitar o efeito de *overffiting* [10], pois tornam o conjunto de dados mais diverso (Krizhevsky et al., 2012; Pigou et al., 2014; Simonyan e Zisserman, 2014; Molchanov et al., 2015). Uma forma de aumento dos dados é a espacial, que inclui as operações de translação, espelhamento horizontal e redimensionamento das imagens (Krizhevsky et al., 2012; Simonyan e Zisserman, 2014). Há também o aumento temporal dos dados, que é geralmente aplicado a vídeos e envolve translação temporal, escalonamento da duração da sequência e deformação no domínio do tempo (Pigou et al., 2014; Molchanov et al., 2015).

Neste trabalho, foram aplicadas tanto estratégias temporais quanto espaciais para aumentar os dados de treinamento. Inicialmente tem-se 995 vídeos da base, dos quais 746 são utilizados para o treinamento e 249 para teste, numa proporção de 75%–25% por classe.

Aplica-se, primeiramente, um deslocamento temporal nos 10 quadros obtidos pela sumarização. Isso é feito somando-se um valor aleatório entre 1 e 4 à posição de cada um desses dez quadros de cada vídeo, dobrando o número de vídeos de treino para 1492. Realiza-se também o espelhamento horizontal e *zoom* dos vídeos originais, resultando em um conjunto com 2984 amostras. Portanto, os dados de treinamento da CNN 3D são esses 2984 vídeos e os de teste são os 249 vídeos citados anteriormente. Todos os quadros foram cortados de 1920 × 1080 *pixels* para 1080 × 1080 *pixels* e redimensionados para 224 × 224 *pixels*, com o intuito de remover uma parte do *background* das imagens e reduzir a quantidade de informação a ser processada. Por fim, os dados de treinamento foram subtraídos da média do conjunto, com o objetivo de normalizar os dados e aumentar a velocidade de aprendizagem.

3.2 Arquitetura

Neste estudo de caso, as CNNs 3D foram utilizadas com o objetivo de fornecer uma linha de base para estudos futuros em reconhecimento de sinais de Libras. Essa escolha foi motivada pelos resultados obtidos em estudos anteriores, indicando a habilidade das redes convolucionais tridimensionais em captar características espaço-temporais (Tran et al., 2014).

[5] Técnica utilizada para posicionar uma imagem sobre outra através do anulamento de uma cor sólida, como o verde claro.
[6] Canon EOS Rebel t5i.
[7] Imagens/quadros por segundo (*frames per second*).
[8] 20 sinais × 5 gravações × 10 sinalizadores.
[9] Sinal: Filho, Sinalizador: 4.
[10] Sobreajuste do modelo aos dados.

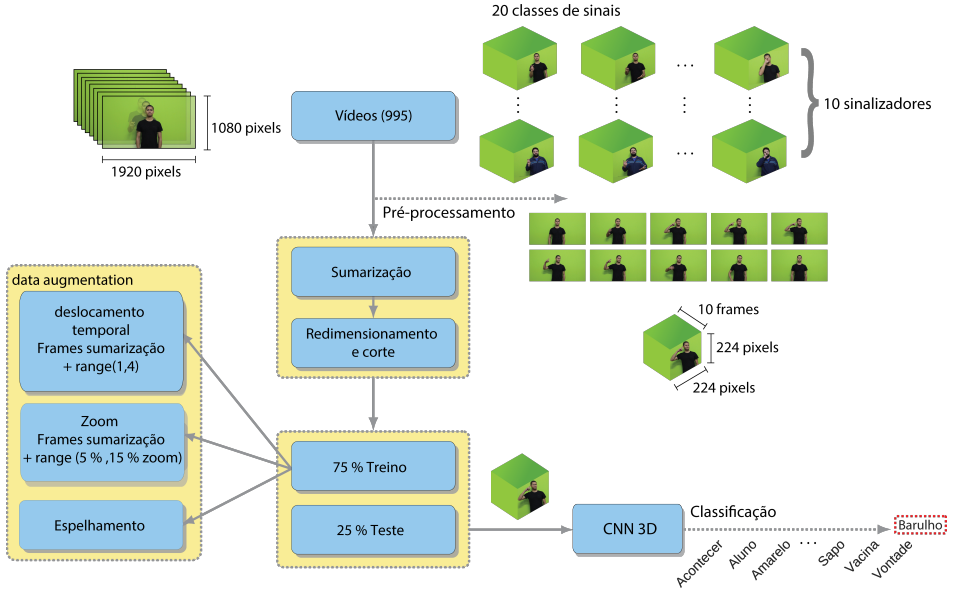


Figura 3. Fluxograma.

A CNN recebe como entrada um volume composto pelos 10 *frames* retornados do pré-processamento, em RGB (dimensões $10 \times 224 \times 224 \times 3$), resultando em um volume de saída que permite que a informação temporal do sinal seja capturada.

A arquitetura da rede consiste em 4 camadas convolucionais, sendo cada uma seguida por uma função de ativação ReLU e uma camada de *max pooling*. As camadas convolucionais possuem 4, 8, 16, e 32 filtros em profundidade, respectivamente. Ao final da rede são utilizadas 2 camadas totalmente conectadas e uma função de ativação *softmax*, que funciona como um classificador. A saída da função de ativação consiste em um vetor contendo a probabilidade de cada um dos 20 sinais corresponderem a uma determinada classe.

Os filtros de convolução utilizados possuem dimensões ($3 \times 3 \times 3$), cujo desempenho em tarefas de análise de vídeos foi avaliado por Tran et al. (2014). As camadas de *max pooling*, que também realizam operações em profundidade, possuem *kernels* ($1 \times 2 \times 2$) e ($2 \times 2 \times 2$), na primeira e demais camadas, respectivamente. A arquitetura da rede é sintetizada na Tabela 2.

Para o treinamento da rede, foi definida uma taxa de aprendizado inicial de 0,001, que foi ajustada conforme o desempenho da mesma. Assim, caso a perda de validação não apresentasse uma melhoria após 3 épocas consecutivas, a taxa foi reduzida por um fator de 10. Foram utilizados lotes de tamanho 128 e o treinamento foi interrompido após 50 épocas. Para evitar *overfitting*, foi empregado o

Tabela 2. Arquitetura.

Descrição	Saída
Volume de entrada	10 × 224 × 224 × 3
Conv3D	8 × 222 × 222 × 4
MaxPool3D	8 × 111 × 111 × 4
Conv3D	8 × 111 × 111 × 8
MaxPool3D	4 × 55 × 55 × 8
Conv3D	4 × 55 × 55 × 16
MaxPool3D	2 × 27 × 27 × 16
Conv3D	2 × 27 × 27 × 32
MaxPool3D	1 × 13 × 13 × 32
Flatten	5408
Fully Connected	128
Dropout	128
Fully Connected	20

Dropout, uma técnica de regularização que elimina alguns neurônios ocultos da rede temporariamente.

O experimento foi realizado 10 vezes, sendo obtidas as métricas de desempenho médias.

4. RESULTADOS E DISCUSSÕES

A figura 4 apresenta a matriz de confusão obtida após 10 iterações do algoritmo de classificação utilizando a CNN 3D. Foram 2984 amostras de treinamento e 249 de teste, alcançando um resultado médio de 72,6% de acerto. Essas iterações garantem uma aleatoriedade no conjunto de treino e de teste, fazendo com que ora a amostra participe do grupo de treinamento, ora do grupo de teste.

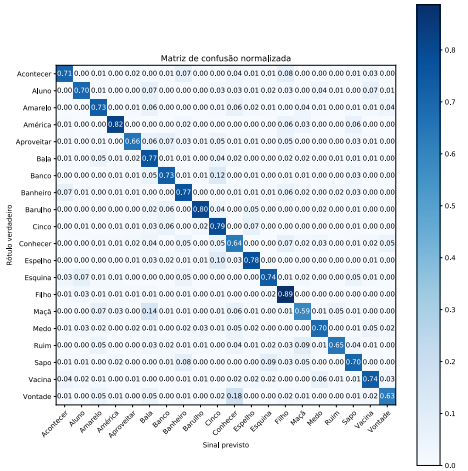


Figura 4. Matriz de confusão normalizada obtida pela média de 10 iterações.

Entre os sinais que apresentaram a menor acurácia, estão "vontade" e "maçã". Observou-se que, em média, 18 % das observações referentes ao sinal "vontade" foram classificadas erroneamente como "conhecer". Esses sinais são representados nas figuras 5 e 6, respectivamente. Percebe-se que, em ambos os casos, o ponto de articulação é o mesmo, isto é, na região em torno do queixo. Assim, sugere-se que a rede neural convolucional foi capaz de aprender as representações relativas ao movimento, mas ainda seria necessário captar outros parâmetros, como a configuração das mãos, para conseguir distinguir entre esses dois sinais. O mesmo acontece com o sinal "maçã" (Figura 7), nas quais 14% das observações foram confundidas com o sinal "bala" (Figura 8). Contudo, conforme a Figura 9, considerando-se os três melhores resultados em cada classe, observa-se que 88,4% das observações do sinal "vontade" e 86,4% do sinal "maçã" apresentaram alta probabilidade de serem corretamente identificados em relação aos seus respectivos rótulos.

Figura 5. Sinal: Vontade.

Figura 6. Sinal: Conhecer.

Vale ressaltar, ainda, que outras ferramentas de pré-processamento podem melhorar o resultado obtido, bem

Figura 7. Sinal: Maçã.

Figura 8. Sinal: Bala.

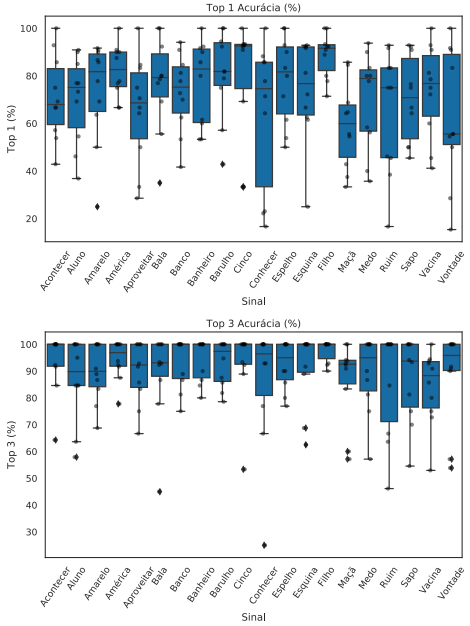


Figura 9. Top 1 e Top 3.

como outra arquitetura de CNN, ou a utilização de uma LSTM. Aplicando a topologia proposta, obteve-se resultados iniciais (Figura 9) com a base de dados a ser disponibilizada para a comunidade científica, com o intuito de ampliar aplicações na área de Libras e abrir espaço para a criação de um tradutor automático da língua.

5. CONCLUSÕES

Este trabalho apresentou um novo conjunto de dados de vídeos para aplicações em reconhecimento da Língua Brasileira de Sinais. Utilizou-se uma CNN 3D para capturar informações inter-quadros e estabelecer um patamar de comparação para trabalhos futuros que apliquem seus métodos na base disponibilizada. Acredita-se que a base de dados apresentada possa contribuir de forma expressiva para o desenvolvimento de novas aplicações em reconhecimento de sinais de Libras e temas correlatos.

REFERÊNCIAS

Almeida, S.G.M. (2014). *Extração de Características em Reconhecimento de Parâmetros Fonológicos da Língua Brasileira de Sinais utilizando Sensores RGB-D*. Ph.D. thesis, Universidade Federal de Minas Gerais, Programa de Pós Graduação em Engenharia Elétrica, Belo Horizonte, Minas Gerais, Brasil.

Almeida, S.G.M., Guimarães, F.G., e Ramírez, J.A. (2015). Um método para sumarização de vídeos baseado no problema da diversidade máxima e em algoritmos evolucionários. In *XII Simpósio Brasileiro de Automação Inteligente (SBAI)*, 1298 – 1303. Natal, Rio Grande do Norte, Brasil.

Almeida, S.G.M., Guimarães, F.G., Rezende, T.M., Almeida, G.T.B., e Toffolo, A.C.R. (2019). Libras-20 dataset. `https://doi.org/10.5281/zenodo.2667329`. doi:10.5281/zenodo.2667329.

Almeida, S.G.M., Guimarães, F.G., e Ramírez, J.A. (2014). Feature extraction in brazilian sign language recognition based on phonological structure and using rgb-d sensors. *Expert Systems with Applications*, 41(16), 7259–7271.

Amaral, L., Júnior, G.L.N., Vieira, T., e Vieira, T. (2019). Evaluating deep models for dynamic brazilian sign language recognition. In R. Vera-Rodriguez, J. Fierrez, e A. Morales (eds.), *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, 930–937. Springer International Publishing, Cham.

Athitsos, V., Neidle, C., e Sclaroff, S. (2008). American sign language lexicon video dataset (asllvd). URL `http://vlm1.uta.edu/~athitsos/asl_lexicon/`.

Capovilla, F.C. (2017). *Dicionário da Língua de Sinais do Brasil. A Libras em Suas Mãos - 3 Volumes*. Edusp.

Conly, C., Doliotis, P., Jangyodsuk, P., Alonzo, R., e Athitsos, V. (2013). Toward a 3d body part detection video dataset and hand tracking benchmark. In *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '13, 2:1–2:6. ACM, New York, NY, USA. doi:10.1145/2504335.2504337. URL `http://doi.acm.org/10.1145/2504335.2504337`.

Escobedo-Cardenas, E. e Camara-Chavez, G. (2015). A robust gesture recognition using hand local data and skeleton trajectory. In *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE. doi:10.1109/icip.2015.7350998. URL `https://doi.org/10.1109/icip.2015.7350998`.

Filho, C.F.F.C., de Souza, R.S., dos Santos, J.R., dos Santos, B.L., e Costa, M.G.F. (2017). A fully automatic method for recognizing hand configurations of brazilian sign language. *Research on Biomedical Engineering*, 33(1), 78–89. doi:10.1590/2446-4740.03816. URL `https://doi.org/10.1590/2446-4740.03816`.

John, V., Boyali, A., Mita, S., Imanishi, M., e Sanma, N. (2016). Deep learning-based fast hand gesture recognition using representative frames. In *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE. doi:10.1109/dicta.2016.7797030. URL `https://doi.org/10.1109/dicta.2016.7797030`.

Kawamoto, A., Bertolini, D., e Barreto, M. (2018). A dataset for electromyography-based dactylology recognition. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE. doi:10.1109/smc.2018.00408. URL `https://doi.org/10.1109/smc.2018.00408`.

Krizhevsky, A., Sutskever, I., e Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

Kumar, P., Gauba, H., Roy, P.P., e Dogra, D.P. (2017). Coupled hmm-based multi-sensor data fusion for sign language recognition. *Pattern Recognition Letters*, 86, 1–8.

Kuo, C.C., Glover, F., e Dhir, K.S. (1993). Analyzing and modeling the maximum diversity problem by zero-one programing. *Decision Sciences*, 24(6), 1171–1185.

Li, W. (2017). Webpage of dr wanqing li. URL `http://www.uow.edu.au/~wanqing/#MSRAction3DDatasets`.

Molchanov, P., Gupta, S., Kim, K., e Kautz, J. (2015). Hand gesture recognition with 3d convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 1–7.

Pigou, L., Dieleman, S., Kindermans, P.J., e Schrauwen, B. (2014). Sign language recognition using convolutional neural networks. In *European Conference on Computer Vision*, 572–578. Springer.

Rautaray, S.S. e Agrawal, A. (2015). Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1), 1–54.

Rawat, W. e Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9), 2352–2449.

Rezende, T.M. (2016). *Aplicação de Técnicas de Inteligência Computacional para Análise da Expressão Facial em Reconhecimento de Sinais de Libras*. Master's thesis, Universidade Federal de Minas Gerais, Programa de Pós Graduação em Engenharia Elétrica, Belo Horizonte, Minas Gerais, Brasil.

Ruffieux, S., Lalanne, D., Mugellini, E., e Khaled, O.A. (2014). A survey of datasets for human gesture recognition. In *International Conference on Human-Computer Interaction*, 337–348. Springer.

Simonyan, K. e Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, 568–576.

Tran, D., Bourdev, L.D., Fergus, R., Torresani, L., e Paluri, M. (2014). C3D: generic features for video analysis. *CoRR*, abs/1412.0767. URL `http://arxiv.org/abs/1412.0767`.

Zhang, L., Zhu, G., Shen, P., Song, J., Afaq Shah, S., e Bennamoun, M. (2017). Learning spatiotemporal features using 3dcnn and convolutional lstm for gesture recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 3120–3128.

C.3 Development and Validation of a Brazilian Sign Language Database for Human Gesture Recognition.

Neural Computing and Applications.

Submissão: Setembro de 2020.

Aceite: Fevereiro de 2021.

DOI: 10.1007/s00521-021-05802-4

Neural Computing and Applications
https://doi.org/10.1007/s00521-021-05802-4

ORIGINAL ARTICLE

Development and validation of a Brazilian sign language database for human gesture recognition

Tamires Martins Rezende[1] · Sílvia Grasiella Moreira Almeida[2] · Frederico Gadelha Guimarães[3]

Received: 10 September 2020 / Accepted: 5 February 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd. part of Springer Nature 2021

Abstract
Sign language recognition is considered the most important and challenging application in gesture recognition, involving the fields of pattern recognition, machine learning and computer vision. This is mainly due to the complex visual–gestural nature of sign languages and the availability of few databases and studies related to automatic recognition. This work presents the development and validation of a Brazilian sign language (Libras) public database. The recording protocol describes (1) the chosen signs, (2) the signaller characteristics, (3) the sensors and software used for video acquisition, (4) the recording scenario and (5) the data structure. Provided that these steps are well defined, a database with more than 1000 videos of 20 Libras signs recorded by twelve different people is created using an RGB-D sensor and an RGB camera. Each sign was recorded five times by each signaller. This corresponds to a database with 1200 samples of the following data: (1) RGB video frames, (2) depth, (3) body points and (4) face information. Some approaches using deep learning-based models were applied to classify these signs based on 3D and 2D convolutional neural networks. The best result shows an average accuracy of 93.3%. This paper presents an important contribution for the research community by providing a publicly available sign language dataset and baseline results for comparison.

Keywords Brazilian sign language · Database · MINDS-Libras · Sign language recognition · Deep learning · CNN

1 Introduction

Gesture recognition is an area in computer vision and machine learning that aims to analyze visual data and develop the machine's ability to understand them [34]. According to [34], there are studies, since the 1980s, reporting static gesture recognition tasks [28], body part detection [25], actions and activities recognition [65] and sign language recognition (SLR). In the gesture recognition hierarchy, as presented by [31], SLR is considered the most important gesture types to be recognized. It is because automatic sign recognition is still a research in development that includes important areas such as computer vision, neural networks, pattern recognition and machine learning [31].

According to the World Health Organization, there are 466 million people with hearing loss (5% of the world population), being 93% of them adults and 7% children [48]. Sign language is a way of communication with these people and among themselves, characterized by visual–gestural communication. Likely to oral languages, sign languages are unique to each culture, with their own grammatical structures. Its basic unit, called "sign", comprises hand configuration, location and movements, palm orientation, face and torso movement, which are manual and non-manual parameters, as illustrated in Fig. 1.

Each sign can be formed by five phonological parameters: hand configuration (HC), articulation point (AP), movement (M), palm orientation (PO) and non-manual expressions (NME). HC refers to the form that the hands take during the execution of the sign. According to [92], there are 19 different HC that can be executed by one or two hands, but this number is not consolidated and there is a variation of this value for each language. In the Brazilian sign language (Libras), for example, Felipe [38] describes 64 HC. AP is the area of the body or space in which the sign is performed. It is divided into head, trunk, arm, hand and neutral space. *M* refers to the internal movements of the hand, wrist and directional movements in space. The last manual parameter, PO, is the palm hand direction during the execution of the sign, being: up, down, inward (towards the body), outward (forward), right and left. Finally, non-manual expressions refer to the movement of the face, head or trunk. They are grammatical elements that perform different functions in the language, such as marking interrogative phrases. The sign illustrated in Fig. 1 demonstrates how the *hurricane* sign is executed. In this case, the hand configuration takes the form of the letter "D", the articulation point is at the side of the body, the movement performed by the hand is spiral upwards with a palm oriented to the left and the facial expression characterized by a frown.

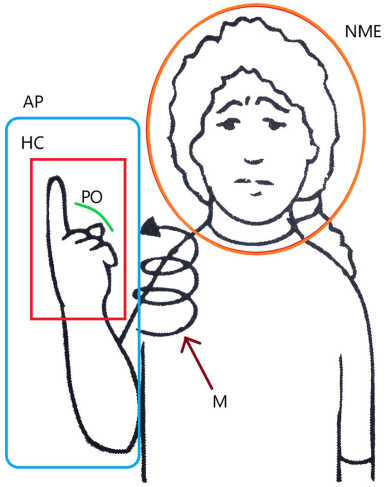


Fig. 1 Phonological parameters of sign language: articulation point (AP), hand configuration (HC), movement (M), palm orientation (PO) and non-manual expressions (NME). Sign: *hurricane* (color figure online). Adapted from [50]

This language structure was defined by [92] in 1960 and it is used for all the 142 languages existing in the world [91], as shown in Fig. 2. These sign languages are often used in computer systems for automatic gesture recognition. Any research in SLR, mainly in Libras, begins with the availability of a representative/robust database that enables the exploration of new approaches. A database is a set of structured information that relates to itself (numbers, images or videos). Previously, in [5, 35, 39], for recording data of Libras signs, the authors have used multimodal sensors such as RGB-D (Red, Green, Blue and Depth). Others have addressed the problem through gloves and wearable sensors [59]. Although these approaches bring scientific contributions, the databases were created for specific purposes and do not allow the reproducibility of the research.

In Brazil, Libras (*LIngua BRAsileira de Sinais*) is the second official language, since the publication of the Law 10,436/2002. According to [5], there are more than 10,000 signs in Libras, making unfeasible to create a database that large, because the recording process is exhaustive for a signaller (who performs the sign) and the data processing is computationally expensive. Besides that, it is necessary to consider the order of magnitude of the stored data which is around gigabyte to terabyte. Thus, in order to fulfil these gaps, we aimed the creation of a Libras' sign database, called MINDS-Libras, through a consistent and well-defined recording protocol. In this paper, we describe how this database was created, reporting technologies used and the main steps in the methodology. Twenty signs were chosen and recorded five times by twelve different people. MINDS-Libras is publicly available at the [70]. We used an RGB camera (Canon EOS Rebel t5) and an RGB-D

Fig. 2 Sign language in the world: 142 languages distributed in 103 countries (color figure online)

✉ Tamires Martins Rezende
tamiresrezende@ufmg.br

Sílvia Grasiella Moreira Almeida
silvia.almeida@ifmg.edu.br

Frederico Gadelha Guimarães
fredericoguimaraes@ufmg.br;
https://minds.eng.ufmg.br/

[1] Graduate Program in Electrical Engineering, Universidade Federal de Minas Gerais, Av. Antônio Carlos 6627, Belo Horizonte, MG 31270-901, Brazil

[2] Department of Industrial Automation, Federal Institute of Minas Gerais - Ouro Preto, Pandiá Calógeras Street, 898, Bauxita, Ouro Preto, MG 35400-000, Brazil

[3] Machine Intelligence and Data Science (Minds) Laboratory, School of Engineering, Federal University of Minas Gerais, Av. Antônio Carlos 6627, Block I Room 2200, Belo Horizonte, MG 31270-901, Brazil

Published online: 01 March 2021

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

sensor (Kinect v2 for Xbox One) for allowing the availability of data in RGB, depth and coordinates of body points. The main concern when building the base was to allow it to be reproducible and expandable for new researchers.

In addition to releasing the MINDS-Libras, we also show a case study using deep learning (DL) approach. In a previous work [23], a 3D convolution neural network (CNN3D) [76, 105] was applied for a part of MINDS-Libras classification (using data from the RGB camera). The architecture used achieved 72% of accuracy, but it was very time-consuming. From this point on, we develop a new DL-based model that outperformed these results. We implemented ten approaches varying the input data, the pre-processing of the data, normalization of the test data and the classification model. A CNN3D was employed with the aforementioned RGB camera data achieving an average accuracy of 93.3% and with lesser time processing.

The main contributions and findings are listed below:

- Available Libras database containing recordings of 20 signs in Libras;
- Presentation of a recording protocol that allows its replicability and reproducibility;
- Analysis of the main features for the construction of a representative sign language database;
- Sign classification methodologies using the data available.

The rest of the paper is organized as follows: in Sect. 2 we briefly review the previous works focused on the creation of sign language databases which are available in the literature. In Sect. 3, a full description regarding the database creation procedures is given. Detailed information on the experimental process for classifying the signs in the DL-based experiment is given in Sect. 4. In Sect. 5, a comparison of the proposed methodology with the baseline is presented. Finally, Sect. 6 concludes our work.

2 Related works

Studies about SLR began around the 1980s. One of the seminal works on this subject was done by [93] and shows tendencies used until today: (1) use of manual parameters to sign recognition and (2) a methodology based on data acquisition, pre-processing, feature extraction and classification [46]. The phonological parameters done by [93] to extract the features used in the classification phase are manual information, such as in [1, 14, 15, 24, 83, 84, 100]. However, it is important to mention that to interpret a sign it is necessary to analyze body movement and also facial expressions [93]. The second tendency refers to a basic step by step process for machine learning tasks. There is a varied number of available techniques that can be employed. In the academic field, the approach will depend on the computational complexity, as well as on the processing time and memory requirements.

The data acquisition is generally done using two different techniques: wearable tools or computer vision, as illustrated in Fig. 3. In the first technique, gloves or sensors are attached to the hands and/or forearms. They are used to capture hands trajectory or manual configuration such as hands shape, finger positions and palm orientation [55, 57, 62, 75, 81, 97]. In computer vision, the systems are independent of the sensors attached to the body but dependent on texture analysis. Some of them can be seen in [10, 29, 32, 54, 60, 82, 95]. Despite being independent techniques, in [47] both were used. The hands' information was analyzed separately and the authors concluded that the data from sensors are more accurate. However, they also pointed out that the use of gloves are more expensive, invasive and often impracticable because it demands touchable actions. The direction of this work is followed by the authors' statement regarding the feasibility of the machine vision over wearable tools.

Given that retrospective, some works have used databases from public repositories such as Kaggle and UCI Machine Learning Repository. They allow the scientific community to use the information to pattern recognition, computer vision, machine learning, among many other computational areas. However, in the SLR field, there is a need for a consolidated dataset in some specific languages as reported by [11, 14, 100, 102]. In these cases, the authors often use either databases from other languages [55, 77], or gesture databases [16, 17, 33, 65, 68, 103] or create their own data to perform the tests.

Among the works that create their own databases, some of them are publicly available, as listed in Table 1. The main differences are related with the data type (gray scale, RGB, RGB-D, points of the face and skeleton), recording process (sequential or isolated), region of interest (body, upper body or hands), scenario (neutral, removed or varied) and the number of signallers. The information contained in these variables allows the scientific community to use

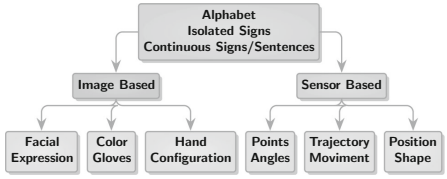


Fig. 3 Possible approaches to sign language recognition: based on images/videos and sensors located in the hands

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 1 Sign language databases

Database	Year	Language	# Samples	Used by	Classifier
Auslan [56]	2002	Australian	2565	[106]	Support vector machines (SVM)
RWTH BONTON-50 [104]	2005	American	483	[67]	Covariance matrix
SIGNUM [99]	2008	German	33,210	[27]	Deep learning
ASLLVD 2008 [12]	2008	American	9800	[13]	Motion energy images
eNTERFACE'06 [9]	2009	American	760	[8]	SVM
MSR Action 3D [64]	2012	American	336	[65]	Dissimilarity
A3LIS [36]	2012	Italian	147	[37]	Hidden Markov model (HMM)
RWTH PHOENIX Weather [40]	2012	German	45,760	[61]	CNN/HMM
RGB-D ASL [26]	2013	American	1113	[26]	Euclidean distance similarity
PSL ToF 84 [80]	2013	Persian	1680	[78]	Dynamic time warping (DTW)
PSL Kinect 30 [79]	2013	Persian	300	[58]	DTW and HMM
Facial expressions [41]	2014	Brazilian	225	[42]	Multilayer perceptron
Libras-34 Dataset (Kinect v1) [4]	2014	Brazilian	170	[3, 5]	SVM
LSA 64 [88]	2016	Argentina	3200	[72]	CNN/recurrent neural network
Libras-10 [7]	2016	Brazilian	100	[43, 86, 87]	SVM/k nearest neighbors
DEVISIGN-D [101]	2016	Chinese	6000	[66, 101]	SVM/deep learning
SLR Dataset [73]	2016	Chinese	25,000	[44, 66]	Deep learning
IITR Sign Language Thermal [71]	2018	Indian	1039	[52]	Kernel-based extreme learning machine
MINDS-Libras [70]	**2019**	**Brazilian**	**1200**	[23]	**CNN 3D**
RKS-PERSIANSIGN [84]	2020	Persian	10,000	[84]	CNN/Long short-term memory
UFOP-Libras [21]	2020	Brazilian	2800	[21]	SVM

Bold indicates the database developed in this paper

different classification methodologies. Each one was created for a specific purpose with different characteristics.

Analyzing the works based on Libras recognition, [5] used data from [4] that contains 34 Libras signs. The phonological parameters (articulation point, hand configuration, movement, palm orientation and non-manual expressions) were analyzed separately and, at the end, the authors employed an ensemble using support vector machines (SVM). An accuracy above 80% on average was achieved with this strategy. In the same way, Cardenas and Chavez [21] created a database containing 56 signs combining information from trajectories, hand and body configurations, using one or both hands. By performing the integration of various features, the aggregation fusion achieved 63.45% of accuracy with linear SVM. In this case, the goal was to verify the efficiency of the different feature extractors.

In [43, 86, 87], the authors used non-manual parameters based on facial expressions using the data presented in [7]. The first used pattern recognition techniques to classify 10 Libras signs. Even with a structured methodology, Rezende [86] pointed out that facial features are not enough for a sign classification, and they suggested using this information merged with manual parameters, as in [2, 31, 37, 45, 49, 51, 53, 60, 63, 74, 85, 98]. Considering facial expressions, emotions recognition is more suitable because it is based only in non-manual parameters. This analysis was performed, for instance, in [43] that used the aforementioned database to recognize the emotions in each sign. Values of average accuracy ranging from 89% to 96% were achieved in 9 experiments. Besides that, it is also possible to analyze grammatically (question, negation, statement) these data as done in [42]. By using their own database, the authors used Kinect to capture the 17 face coordinates (x, y, z).

It can be seen that the approaches used in the literature to perform SLR tasks are as varied as possible and there is still no specific technique or tool to solve this kind of problem. Thus, many criteria need to be analyzed, such as the grammar of each language, phonological parameters, database, feature extraction, and the classification method. Focusing on the database as a whole, we listed in Table 2 some desirable characteristics that have already been observed in the literature combined with our own ideas. This analysis takes into consideration the necessary requirements to cover most applications in machine learning tasks in SLR.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 2 Desirable characteristics for a sign language database

Characteristics	Description	References
Phonological parameter	The sign is composed of facial and manual information, then it is desirable that the database contains both data. Regardless of the system's objective, this database is going to be complete enough and can be used in various applications	[21, 22, 30, 64, 70, 79]
Sign nature	There are static and dynamic signs, which compose the sign language. Creating a database with these data type demonstrates that the base is representative. Most works developed their datasets with only one of these	[4, 70, 80]
Sign type	A database containing signs of the alphabet, isolated signs and sentences (Fig. 3) can be considered complete because it permits greater representation of the vocabulary of the language	-
Size	The database size is a subjective variable because it is not simple to record signs on a large scale. It is desirable that researchers contribute with recordings, such that the recording is made constantly and uninterruptedly	[12, 40, 70, 88]
Number of signallers	The base should have more than one signaller executing the sign. This means that the system, when based on images, does not specialize in the person's physical characteristics. Also, small variations in sign execution can occur when it is executed by different signallers. Independent of the data acquisition system, it is important that the system is prepared for these changes	[26, 40, 64, 70, 88]
Recording scenario	The lighting control is relevant so that the recordings are made with quality. However, if the system is used in public environments, the base should be recorded with some technology that permits the insertion of variable backgrounds. Thus, the classifier can learn different scenarios and distinguish the executed sign	[4, 7, 70]
Recording protocol	Following a recording protocol is the assurance of a standard in the recordings and permits other researchers to contribute to the evolution of the research, facilitating the sampling of the data	[4, 7, 70]

3 MINDS-Libras database

This section presents the main steps of creating the Brazilian sign language database, called MINDS-Libras, following a recording protocol allowing to be replicated and, mainly, reproducible. MINDS-Libras has most of the desirable characteristics listed in Table 2 for a sign dataset and it is available publicly in [70]. MINDS-Libras aims to cover the existing gap of a standardized database that poses different challenges to the researchers in artificial intelligence (AI).

In this work, the recording protocol definition follows the main principles described in [3, 86, 89] and present: (1) chosen signs, (2) signaller characteristics, (3) sensors and software used for the video acquisition, (4) recording scenario and (5) data structure.

Initially, twenty signs were selected by a Libras expert. This choice was based on the diversity of the aforementioned phonological parameters. Each sign was recorded five times by 12 signallers, totalling 1200 samples. The signallers were men and women, deaf and hearing, without distinction neither of the clothes nor of the clothes' colors that they were wearing. During the recording process of each sign, it was requested to the signaller to look straight to the camera with neutral facial expression. Her/his position was fixed at the center of the video and with both hands down at the side of the legs, registering the beginning and the end of each sign sample. In the recording studio, which is a controlled environment, the sensors had fixed positions, in a way that they received the same amount of illumination. The recorded angle captured the facial expression and manual movements. To provide the videos in mp4 format with depth information, we used at the same time a professional digital camera Canon EOS Rebel t5i and the RGB-D Kinect v2 for Xbox One sensor.

3.1 Signs

In [18–20], the authors documented more than 10,000 Libras signs which enable communication just like in any language. However, the visual ones involve subtleties often perceptible by humans within a context that it is not trivial for AI. This specificity of sign languages requires the previous definition of some parameters for database creation.

Because of this inherent complexity, the task of choosing the signs for a database creation naturally falls to a language specialist. For this reason, in this paper, a Libras teacher selected 20 signs based on phonological parameters diversity. They are: *To happen*, *Student*, *Yellow*, *America*, *To enjoy*, *Candy*, *Bank*, *Bathroom*, *Noise*, *Five*, *To know*, *Mirror*, *Corner*, *Son*, *Apple*, *Fear*, *Bad*, *Frog*, *Vaccine* and *Will*, see Fig. 4. Since the facial expression is strongly related to emotion/sentiment of a sign, some signs may not present this feature. Regarding to the movement, in MINDS-Libras, most of signs are dynamic, except *America* (Fig. 4d) and *five* (Fig. 4j).

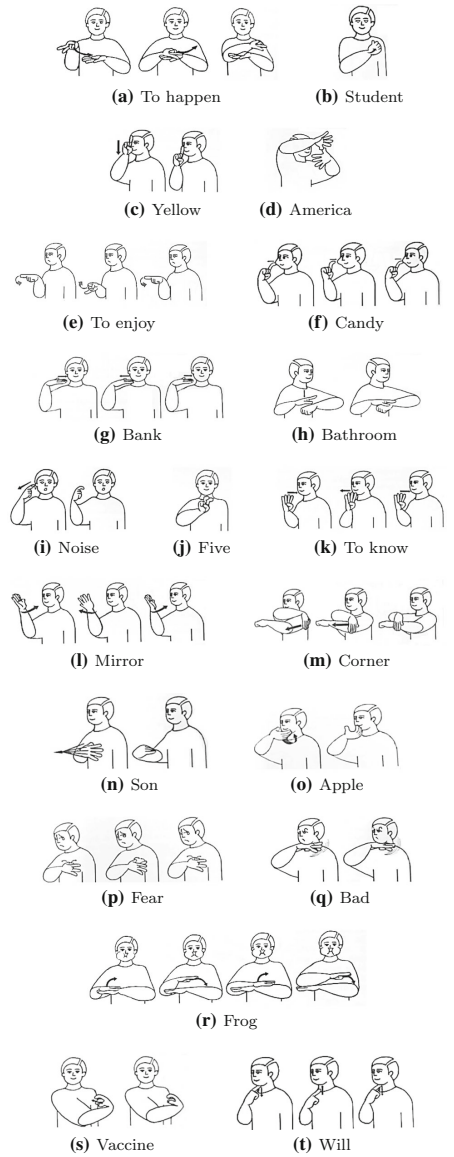


Fig. 4 Signs that make up the MINDS-Libras database. *Source* [18–20]

3.2 Signallers

Different signallers may decrease the bias present in the data. MINDS-Libras signs were executed by 12 signallers, varying sex, age and fluency in Libras, as listed in Table 3. Although the clothing colors were not standardized, most of them are black, as illustrate Fig. 5. In the end, each signaller was recorded 100 times (20 signs $\times$ 5 repetitions). To know, this process took about 2 hours per signaller.

3.3 Sensors and software

To capture the Libras signs, an RGB camera (Canon EOS Rebel t5i) and the RGB-D sensor (Kinect v2 for Xbox One) were used. The data available are the RGB videos with 1920 $\times$ 1080 resolution. Moreover, the RGB-D sensor provides signaller's body coordinates in a text file format and depth videos with 640 $\times$ 480 resolution.

Many sensors can be used to record this type of data. In this case, we used a RGB camera due to its ability to capture the scene in high resolution and, also, the RGB-D sensor for the additional information. The recordings were made simultaneously by using both sensors.

In order to reduce the processing time, the RGB camera remained on while recording the five repetitions of each sign. Further, since only one video was captured, a preprocessing was performed to divide the samples per sign. Another important issue is related to the data recorded by the RGB-D sensor. To obtain synchronously all the data, we invoked a Kin2 library [94] from MATLAB Student (Version: 2018). Since the Kinect needs a short time to correctly identify this information, we have implemented an interface that allows to start recording after a command given by the user. Then, five seconds of sign execution are recorded, totalling 150 frames (30 frames per second). This time was necessary to avoid error/crash system due to the storage of data in memory. We consider it is enough to record the signs.

3.4 Recording studio

The recording studio was built to allow adding or removing different backgrounds in the videos in a way to explore the performance of algorithms based on visual patterns. For this, the scenario had lighting and fixed background of Chrome Key, as illustrated in Fig. 6. The distance between the signaller and the RGB-D sensor was $\approx$ 1.60 m and

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 3 Signallers characteristics

Signaller	Sex	Age	Previous knowledge	Skin color	Clothes
1	Male	30–40	Fluent (deaf)	Brown	Blue
2	Male	20–30	Fluent	Brown	Black
3	Female	20–30	Intermediate	White	Black
4	Female	30–40	Fluent (teacher)	Brown	Wine
5	Female	30–40	Intermediate	Brown	Black
6	Female	30–40	Fluent	Black	White
7	Female	20–30	Basic	Brown	Black
8	Female	40–50	Basic	White	Black
9	Male	20–30	Intermediate	White	Black
10	Female	20–30	Intermediate	Brown	Black
11	Female	20–30	Intermediate	White	Black
12	Male	20–30	Intermediate	Brown	White

(a) Signaller 1 **(b)** Signaller 2 **(c)** Signaller 3 **(d)** Signaller 4 **(e)** Signaller 5 **(f)** Signaller 6

(g) Signaller 7 **(h)** Signaller 8 **(i)** Signaller 9 **(j)** Signaller 10 **(k)** Signaller 11 **(l)** Signaller 12

Fig. 5 Signallers that participated in the recording of the MINDS-Libras database

Fig. 6 MINDS-Libras database recording scenario (indoor)

$\approx$ 1.92 m for the RGB camera. The angle seen by the sensors comprises the movement of the upper limbs.

3.5 Data

At the end of the recording process, there is data related to the RGB-D sensor and RGB camera. Unfortunately, some data from RGB camera was lost and it is described below.

Fig. 7 Example of a frame captured from the RGB camera

This may be due to the camera going into sleep mode or battery issues.

- Signaller 3: *Student*, *America* and *Five*;
- Signaller 4: *Son*;
- Signaller 9: *Yellow*, *Bathroom*, *To know*, *Corner* and *Fear*.

Due to these losses, from the 1200 samples initially planned, 1155 recordings (= 1200 − 5 recordings × 9 signs) are available in [70]. The entire database has

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

64.8 GB, and an example of a captured frame is illustrated in Fig. 7.

The RGB-D sensor data have ".mat" extension, i.e. MATLAB format file. However, to ensure that the use of this data is not restricted to researchers using this software, we have chosen to make available the RGB and depth in ".mp4" format, and the body points and face data in ".txt" file. Figures 8, 9, 10, 11, 12 and 13 illustrate these data. Each data has information about 150 frames. The average size of each video is, approximately, 20 Mb for RGB and 1 Mb for depth. The text file has 654 Kb for body data and 22 Mb for face data.

The structure with all the information recorded by RGB-D sensor is described in Table 4. The files containing the body data were organized as in Table 5. There is one file for each sign recorded with seven different categories of information about the 25 points, as shown in Figs. 10 and 11: (1) Spine Base, (2) Spine Mid, (3) Neck, (4) Head, (5) Shoulder Left, (6) Elbow Left, (7) Wrist Left, (8) Hand Left, (9) Shoulder Right, (10) Elbow Right, (11) Wrist Right, (12) Hand Right, (13) Hip Left, (14) Knee Left, (15) Ankle Left, (16) Foot Left, (17) Hip Right, (18) Knee Right, (19) Ankle Right, (20) Foot Right, (21) Spine Shoulder, (22) Hand Tip Left, (23) Thumb Left, (24) Hand Tip Right and (25) Thumb Right. There are 13 lines (or data) for each frame. This order is repeated sequentially up to 1950 lines (13 lines × 150 frames), representing the sign video. As the recordings made captured the information from the upper body, it is suggested to exclude the points 13–20, resulting in 17 points seeking to represent the most significant information from the sign.

Regarding to the face data, the same organization was adopted, as shown in Table 5. In this case, we have seven categories of information, describing 11 data, distributed in 1650 (11 lines × 150 frames) lines in the ".txt" file. Figs. 13, 14, 15 and 16 exemplify the FaceBox information plotted in an RGB frame, also FaceModel,

Fig. 8 Example of the RGB data of a frame captured from the RGB-D sensor

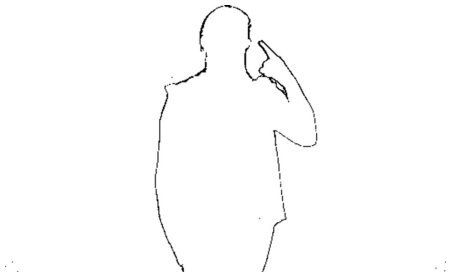


Fig. 9 Example of the depth data of a frame captured from the RGB-D sensor

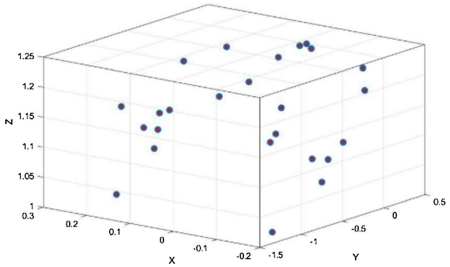


Fig. 10 Example of the *x*–*y*–*z* position data of a frame captured from the RGB-D sensor

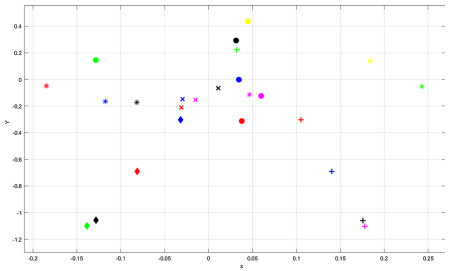


Fig. 11 Example of the *x*–*y* position data of a frame captured from the RGB-D sensor

ColorFaceModel and DepthFaceModel. These last two data are relative values of FaceModel in relation to the RGB frames and depth, respectively.

The MINDS-Libras database tries to cover a part of the literature that either has few available data or does not follow a recording protocol, making the data exploration a very hard task. This section described a replicable and reproducible recording protocol, allowing other researchers

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

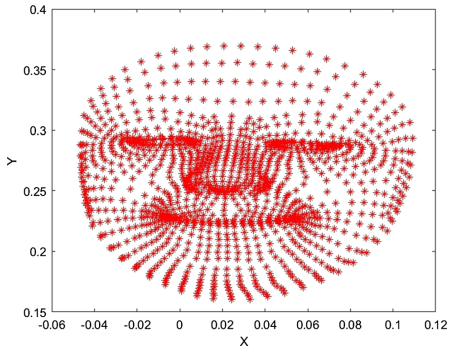


Fig. 12 x–y coordinates of 1347 points that represent the signaller face

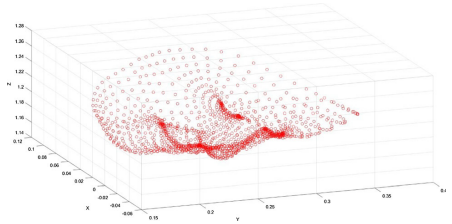


Fig. 13 x–y–z coordinates of 1347 points that represent the signaller face

Table 4 RGB-D sensor data

Data	Struct	Description
Body	[150 × 1 struct]	See Table 5
Color	[1920 × 1080 × 3 × 150 uint8]	RGB videos
Depth	[424 × 512 × 150 uint16]	Depth videos
Face	[150 × 1 struct]	See Table 5
Time	[1 × 150 double]	Defined time between 0 and 5 s, which marks the execution of each of the 150 frames
Gender	Character	'F' if female and 'M' if male
IsDeaf	Character	'Y' if deaf and 'N' if non-deaf

to contribute to increasing the base. Even if they do not publish the sign, expanding the MINDS-Libras, this work provides a direction to follow.

Fig. 14 FaceBox: rectangle plotted on the signaller face

Fig. 15 FaceModel plotted in RGB frame

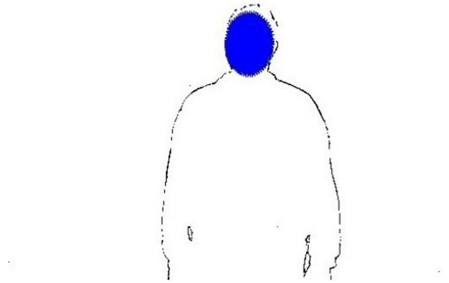


Fig. 16 FaceModel plotted in depth frame

4 Proposed approach

The main goal of this work is to create a publicly available database recorded in a standardized way and to present a robust methodology to classify the signs. Previously, in [23], the authors performed data pre-processing and applied a CNN3D-based architecture for the classification. Since they used part of MINDS-Libras data-set, as detailed in Sect. 4.1, it was considered as our baseline. To improve

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 5 Body and face data file

File	Data	Line	# Columns	Description
Body	Position (Fig. 10)	1	25	x-coordinate of 25 joints
		2	25	y-coordinate of 25 joints
		3	25	z-coordinate of 25 joints
	Orientation	4	25	Orientation in x of 25 joints (pitch)
		5	25	Orientation in x of 25 joints (yaw)
		6	25	Orientation in x of 25 joints (roll)
	TrackingState	7	25	Tracking states of the 25 joints
	LeftHandState	8	1	Left hand state
	RightHandState	9	1	Right hand state
	ColorPosition	10	25	x-Coordinate position in RGB frame
		11	25	y-Coordinate position in RGB frame
	DepthPosition	12	25	x-Coordinate position in depth frame
		13	25	y-Coordinate position in depth frame
Face	FaceBox (Fig. 14)	1	4	x–y coordinates that define a rectangle on the signaller face
	FaceRotation	2	3	x–y–z coordinates of head movement
	HeadPivot	3	3	x–y–z coordinates for reference at the point of the head
	AnimationUnits	4	17	detect and track faces showing the 17 animation units
	FaceModel (Fig. 13)	5	1347	x-coordinates of the face
		6	1347	y-coordinates of the face
		7	1347	z-coordinates of the face
	ColorFaceModel (Fig. 15)	8	1347	x-coordinates of the FaceModel in RGB frame
		9	1347	y-coordinates of the FaceModel in RGB frame
	DepthFaceModel (Fig. 16)	10	1347	x-coordinates of the FaceModel in depth frame
		11	1347	y-coordinates of the FaceModel in depth frame

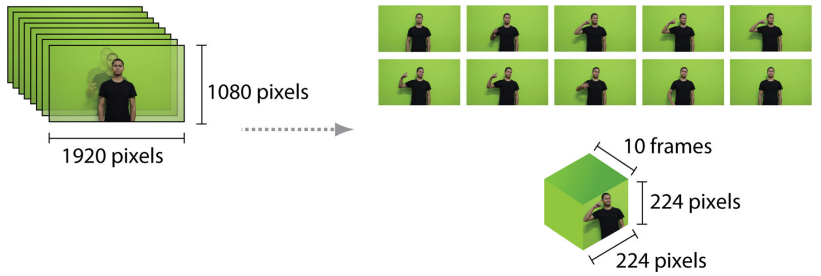


Fig. 17 Frames pre-processing: summarizing and resizing. Adapted from [23]

the results, we made some modifications in the classification structure. The experiment is described in Sect. 4.2.

4.1 Baseline

In [23], the authors used data from RGB camera, without signaller three and nine (995 MINDS-Libras samples). The first step was data pre-processing. A video summarizing technique was applied to eliminate redundant information and standardizing the recordings to the same size. Besides, the video processing is a step that requires large computational resources. The summarizing technique used was based on the maximum diversity problem, implemented by [6], returning 10 frames to represent the sample. Then, all frames were cut from 1920 × 1080 pixels to 1080 × 1080 pixels and resized to 224 × 224 pixels, to remove a

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

portion of the background of the images and reduce the amount of information to be processed. Figure 17 shows the entire process.

The second part of the pre-processing step was the application of three different types of data augmentation in the training set: (1) temporal displacement in the 10 frames obtained by summarizing. This is done by adding a random value between 1 and 4 to the position of each of these 10 frames of each video; (2) horizontal mirroring; and (3) zoom from 5 to 15% on the original frames. This means that the 75% training samples (75% of the 995 = 746) resulted in 2984 samples. The remaining data (25% = 249 samples) were used for testing. Then, both were normalized between [0 and 1]. The normalization of values for the training set considered data from the training set only.

The last step was feature extraction and classification of the signs. CNN3D was chosen due to the ability to capture spatial-time characteristics [96]. It receives as input those 10 frames illustrated in Fig. 17, in RGB (dimensions $10 \times 224 \times 224 \times 3$).

The CNN3D architecture consists of four convolutional layers, each one followed by a ReLU activation function and a max-pooling layer. Also, two fully connected layers and a softmax activation function were used, which worked as a classifier. The output consists of a vector containing the probability that each of the 20 signs corresponds to a certain class. The convolution layers have 4, 8, 16, and 32 filters in depth, respectively, with $3 \times 3 \times 3$ dimensions. Max-pooling have kernels $1 \times 2 \times 2$ in the first layer and $2 \times 2 \times 2$ in the others. This architecture is summarized in Table 6.

For the training of the network, an initial learning rate of 0.001 was defined, which was adjusted according to its performance. Thus, if the validation loss did not improve after three consecutive epochs, the rate was reduced by a factor of 10. Batch of size 128 was used and the method ended after 50 epochs.

Table 6 CNN3D model from baseline, *Experiments 1* to *6* and *Experiments 10*

Description	Output	# Parameters
Input	$10 \times 224 \times 224 \times 3$	–
Conv3D	$8 \times 222 \times 222 \times 4$	328
MaxPool3D	$8 \times 111 \times 111 \times 4$	0
Conv3D	$8 \times 111 \times 111 \times 8$	872
MaxPool3D	$4 \times 55 \times 55 \times 8$	0
Conv3D	$4 \times 55 \times 55 \times 16$	3472
MaxPool3D	$2 \times 27 \times 27 \times 16$	0
Conv3D	$2 \times 27 \times 27 \times 32$	13,856
MaxPool3D	$1 \times 13 \times 13 \times 32$	0
Flatten	5408	0
Fully Connected	128	692,352
Dropout	128	0
Fully Connected	20	2580
Activation	20	0
# Parameters:	713,460	

4.2 Experiments

The architecture described above was expanded here. The input was the RGB camera video: 995 videos, each one containing 10 frames of three channels of 224×224 pixels. From these data, the samples were randomized and partitioned into train (75%) and test (25%) sets, per sign. The first modification was the exclusion of the data augmentation step, called *Experiment 1*. It was necessary due to the large amount of data used which requires high computational cost. Regarding the data normalization, the same procedure applied to the training data by [23] was used in the test set. This modification is important because these data are unknown for the model and may improve the network performance.[1] This defines *Experiment 2*.

Up to here, RGB images were used. Each frame had three channels resulting in images in format $224 \times 224 \times 3$. For the *Experiment 3*, the hypothesis tested was if using grayscale ($224 \times 224 \times 1$) it is possible to reduce the time classification since it reduces from three to one channel while also achieving high accuracy.

The next three experiments were implemented also based on the grayscale images. In *Experiment 4*, the aim was to insert the motion information from the sign by using optical flow proposed by Lucas–Kanade technique [69]. Also, the Shi–Tomasi detection method [90] was employed to identify the points to be traced by the sparse flow algorithm. It creates a new "image" of optical flow, which was passed as a second channel along with the videos in grayscale. In *Experiment 5*, similar to the last one, we use 10 frames 224×224 and two channels as input, but based on the HOG descriptor. This technique is commonly used to detect edges and objects. In our experiment, its application helps in the characterization of the sign that is being executed, without highlighting the physical characteristics of the signaller. Finally, *Experiment 6* was implemented by adding the data augmentation procedures used in [23] to the grayscale images. Despite increasing processing time, the hypothesis to be tested is that by increasing the volume of data, the network improves its performance.

Shortly, in the Experiments from 1 to 6, gradual changes were made so that the best characteristics could be extracted from the videos and thus improving the accuracy in sign recognition. These are summarized and presented in Table 6.

However, RGB-D sensor offers additional data beyond the RGB images. With this, the next experiments use as input the positions of some points of the body for the classification system. Analyzing Fig. 11, there are five points related to the hands (elbow, wrist, hand, hand tip and thumb). In *Experiment 7*, the x–y positions of each one of these points were used with reference to the position of the head, as shown in Eq. (1). These points were organized in a state vector (S), illustrated in Eqs. (2) and (3). Thus, a state represents 1 frame of the sign. As the RGB-D sensor videos have 150 frames, we have 150 states per sample.

$$(x,y)_{pX} = (x,y)_{\text{pointX}} - (x,y)_{\text{head}} \tag{1}$$

where pX is the position of the point$X$ with reference of the *head* position (point 4), $X \in 10, 11, 12, 24, 25$ to right hand and $X \in 6, 7, 8, 22, 23$ to left hand.

$$S_{\text{frame } i,\text{right hand}} = \left[\underbrace{(x_i,y_i)_{p10} \ldots (x_i,y_i)_{p25}}_{\text{coordinates of points 10,11,12,24,25}} \right] \tag{2}$$

$$S_{\text{frame } i,\text{left hand}} = \left[\underbrace{(x_i,y_i)_{p6} \ldots (x_i,y_i)_{p23}}_{\text{coordinates of points 6,7,8,22,23}} \right] \tag{3}$$

Given the state of each frame, the concept of recurrence of dynamic systems was applied. The expectation is that the hand shift pattern throughout the sign execution will be captured by the recurrence matrix and that CNN will be able to identify this pattern for each sign. This matrix (M_R), illustrated by Fig. 18, represents the Euclidean distance (Eq. 4) between the states of each hand, resulting in $150 \times 150 \times 2$ tensor for each sample. Each tensor was standardized from 0 to 1 and the same proportion of train/test was applied (75–25%). However, as in this case, RGB-D sensor data are being used, we have 900 samples for training and 300 for test.

$$\underbrace{\mathrm{d}H_{i,j}}_{\text{frame } i,\text{ frame } j} = \text{dist}(S_i, S_j) \tag{4}$$

$$\begin{cases} \forall i,j \in 1,2,\ldots 150 \\ H = \text{Hand Right}(R) \text{ or Hand Left } (L). \end{cases}$$

With this approach, the CNN3D was replaced by a CNN2D, with a similar arrangement: 4 convolutional layers (4, 8, 16, and 32 filters), each followed by a ReLU activation function and a max pooling layer. Then, 2 fully connected layers and a softmax activation function as a classifier. The convolution filters used have dimensions 4×4 and the max-pooling layers with kernels 2×2. For training, the only change was in the number of epochs, from 50 to 400. The network architecture is summarized in Table 7.

For *Experiment 8*, Eq. (1) is replaced by Eq. (5). The only change made is adding the information from the z-axis of the state vector [Eqs. (2) and (3)] and recalculating the M_R. The aim is to verify the importance of depth points.

$$(x,y,z)_{pX} = (x,y,z)_{\text{pointX}} - (x,y,z)_{\text{head}} \tag{5}$$

From *Experiment 8*, *Experiment 9* was designed implementing data augmentation to training data. In this case, we want to investigate whether the volume of data that CNN2D will receive is significant and can improve the results. Randomly 10 frames received a Gaussian noise with a sigma of 5% of the maximum range of points and a new M_R was calculated. This process was performed five times, increasing the training sample to 5400 samples. For training, the only change was in the number of epochs, from 400 to 100.

Finally, *Experiment 10* was implemented by mixing the best results obtained with the classification of the signs in which the input was image data (*Experiment 6*) and when the hand points were used (*Experiment 9*). An adaptation was made to *Experiment 9*: the same 995 samples from *Experiment 6* were now used to make it possible a fair comparison between the models. Figure 19 presents a scheme of the proposed network architecture. CNN3D was used to classify the sign videos and CNN2D to classify the recurrence matrix, replicating the models presented in Tables 6 and 7, respectively. For the CNN3D, the 10 most significant frames of each sample were used, while for the CNN2D the body point information for all 150 frames was considered. In the latter case, the computational cost for computing the recurrence matrix (M_R) formed by all the points is low, therefore all frames were considered. The sign classification was performed from the weighted average of the predictions of each model. This approach seeks to emphasize the importance of extracting

$$M_R = \begin{bmatrix} 0 & dL_{1,2} & \cdots & dL_{1,150} \\ dL_{2,1} & 0 & \cdots & dL_{2,150} \\ dL_{3,1} & dL_{3,2} & \cdots & dL_{3,150} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & dR_{1,2} & \cdots & dR_{1,150} \\ dR_{2,1} & 0 & \cdots & dR_{2,150} \\ dR_{3,1} & d_{3,2} & \cdots & dR_{3,150} \\ \vdots & \vdots & \ddots & \vdots \\ dR_{149,1} & dR_{149,2} & \cdots & dR_{149,150} \\ dR_{150,1} & dR_{150,2} & \cdots & 0 \end{bmatrix}$$

Fig. 18 Recurrence matrix

[1] See https://stats.stackexchange.com/questions/211436.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 7 CNN2D model from *Experiments 7–10*

Description	Output	# Parameters
Input	$150 \times 150 \times 2$	–
Conv2D	$147 \times 147 \times 4$	132
MaxPool2D	$73 \times 73 \times 4$	0
Conv2D	$73 \times 73 \times 8$	520
MaxPool2D	$36 \times 36 \times 8$	0
Conv2D	$36 \times 36 \times 16$	2064
MaxPool2D	$18 \times 18 \times 16$	0
Conv2D	$18 \times 18 \times 32$	8224
MaxPool2D	$9 \times 9 \times 32$	0
Flatten	2594	0
Fully connected	128	331,904
Dropout	128	0
Fully connected	20	2580
Activation	20	0
# Parameters: 345,424		

characteristics from the video signs, but also to use data that are independent of the physical characteristics of the signallers, such as the points of the hand that describe the trajectory in the movement.

In summary, this section presented ten experiments conducted with data provided by MINDS-Libras. All the experiments were programmed in python, using the Google Colab environment. The most used libraries were NumPy, Pyplot, Sklearn, Keras and TensorFlow. It is possible to see in Fig. 20 the main steps of this paper as well as the techniques applied. This kind of approach is commonly used in pattern recognition and machine learning for providing challenges to the users.

5 Experimental results

This section describes the results for the ten experiments previously detailed by using the new available MINDS-Libras database with different types of information. These results may be used as a guide for new studies, and it allows to understand further directions related to sign language recognition.

Table 8 summarizes the characteristics of each experiment, highlighting the changes that occurred among implementations. Besides, the last two columns show the accuracy average and time, in seconds, spent with classification process (10 iterations). The criterion for comparison among the implementations performed per class in this work was the accuracy (or True Positive (TP)) in the test set, but other metrics were also analyzed. Precision [Eq. (6)] can be used in a situation where false positives (FP) are considered worst than false negatives (FN) and recall [Eq. (7)] for vice versa. In our approach, both metrics are equally important, because a misclassification indicates that the CNN learnt spurious parameters which were not important to recognition.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{7}$$

As can be seen in Table 8, *Experiment 1* achieved 59.7% of average. Since this part did not use data augmentation, the usage of GPU (Graphics Processing Unit) was unnecessary. It was worse than the baseline. *Experiment 2* scored 89.9%, greater than both the first one and the baseline, showing that normalizing the test set by the training data is necessary as aforementioned explained. Analyzing the approaches that used images as input, *Experiment 3* show the second highest accuracy, 91.6%, and the shortest time processing, confirming the hypothesis designed. In the next two experiments, *4* and *5*, we add movement information and border detection. It was good compared to the baseline, but it shows similar accuracy than the one obtained in *Experiment 3*, but with high time consuming. As there were no significant changes when the Optical Flow (*Experiment 4*) and the HOG descriptor (*Experiment 5*) were used, *Experiment 6* was structured from the best result obtained at the time. It finishes the first part of the experiments and confirms a characteristic seen when using CNN: the volume of data is a significant parameter for the classifier. Although the experiments were statistically equivalent, there was an improvement in the classification with an average accuracy of 93.3%.

Experiments 7, *8* and *9* used points of the hand, forming the recurrence matrix as the network input. This concept allows the network to receive data that is independent of the physical characteristics of the signallers. As promising as the previous results were when compared to the baseline, we now have a less costly approach in terms of time classification and that inspires new experiments. Among the results obtained, it was perceived that the depth information was relevant. The increase in data, a characteristic of *Experiment 9*, also proved promising for this data. When compared with experiments that have images as input to the system, these experiments are statistically inferior, but when compared with other works that also used body points, as in [21], the accuracy found is within expectations. The approach performed in *9* added to *Experiment 6* (*Experiment 10*) resulted in a significant improvement in accuracy. The weighted average applied was based on the results obtained separately in each experiment. Thus, the

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

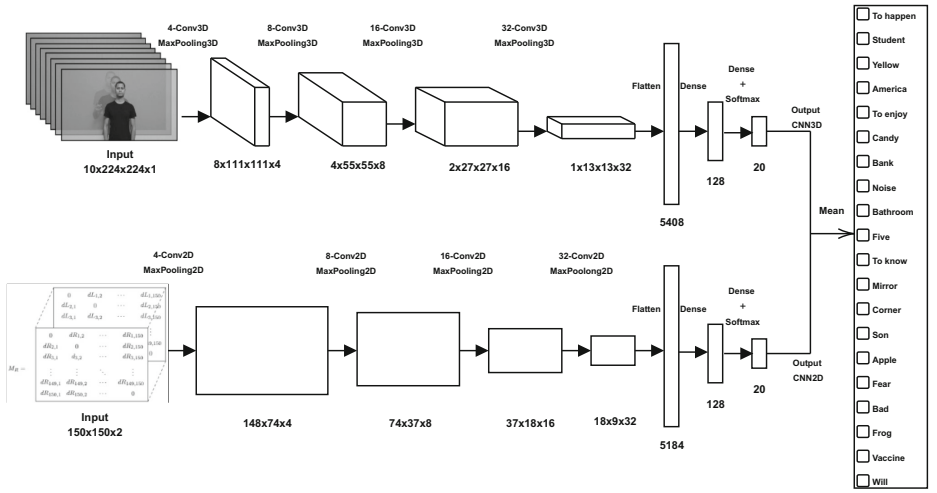


Fig. 19 Proposed network architecture: CNN3D classifying the grayscale images and CNN2D the recurrence matrix obtained by the *x*–*y* coordinates of the hand points

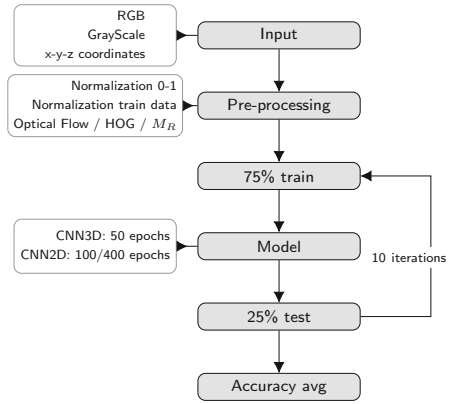


Fig. 20 Experimental flow diagram

weight equal to nine was given for *Experiment 6* and six for *Experiment 9*.

Figure 21 shows a comparison between the experiments, all against all, considering a 95% confidence level. The experiments that have the images as network input, i.e. *Experiments 2, 3, 4, 5, 6* and *10* (Group 1) perform better than *Experiments 1* and *Experiments 7, 8* and *9* (Group 2). The Group 1 experiments are statistically equivalent to each other. For Group 2, this relation prevails, except for 9 is better than 7.

For sake of simplicity, we decided to discuss the main findings of the *Experiment 6* that achieved the best performance regarding to accuracy and time processing. Table 9 detailed the results for the main metrics for assessing network performance in 10 iterations of the classification algorithm based on CNN3D. The values of FN and FP corroborate with the result of accuracy. However, there are three signs that deserve to be highlighted: (1) the sign *To happen* and *To know* had an accuracy and precision of 100%. The samples were classified as these signs is correct and there were no samples of *To happen* and *To know* classified as other signs. (2) Sign *Noise* had 100% of recall. This means that no sign has been confused with the sign *Noise*.

Figure 22 shows the accuracy variance of each sign, in *Experiment 6*. The signs that have less than 90% of accuracy are *Fear*, *Frog*, *Will* and *Corner*. On average, 8% of the observations referring to the sign *Will* were erroneously classified as *To know*, represented in Fig. 4t, k, respectively. It can be noticed that both have the same articulation point in the region around the chin. The same happens with the sign *Frog* (Fig. 4r), in which 9% of the observations were confused with the sign *Corner* (Fig. 4m). In this case, the articulation point and the movement performed by the hand are similar.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 8 Experiments with the MINDS-Libras database

Experiment	Input data	Pre-processing	Test data normalization	Model	Accuracy avg.	Time
Baseline	RGB	S/R and DA	0–1	CNN3D	72.6%	–
1	RGB	↪ − DA	0–1	CNN3D	59.7% (± 20.52)	2734
2	RGB	↪ − DA	↪ + training avg	CNN3D	89.9% (± 1.02)	2802
3	GrayScale	↪ − DA	↪ + training avg	CNN3D	91.6% (± 1.57)	1730
4	GrayScale	↪ + Optical Flow	↪ + training avg	CNN3D	90.5% (± 1.55)	2521
5	GrayScale	↪ + HOG	↪ + training avg	CNN3D	88.8% (± 3.14)	2492
6	GrayScale	↪ + DA	↪ + training avg	CNN3D	93.3% (± 1.69)	5706
7	*x*–*y* position	Recurrence matrix	0–1	CNN2D	50.3% (± 2.69)	618
8	↪ + *z* position	Recurrence matrix	0–1	CNN2D	53.1% (± 2.56)	660
9	*x*–*y*–*z* position	↪ + DA	0–1	CNN2D	61.2% (± 2.02)	932
10	↪ + GrayScale	↪ + S/R and DA	↪ + training avg	↪ + 3D	92.0% (± 3.61)	7111

S/R summarizing and resize, *DA* data augmentation

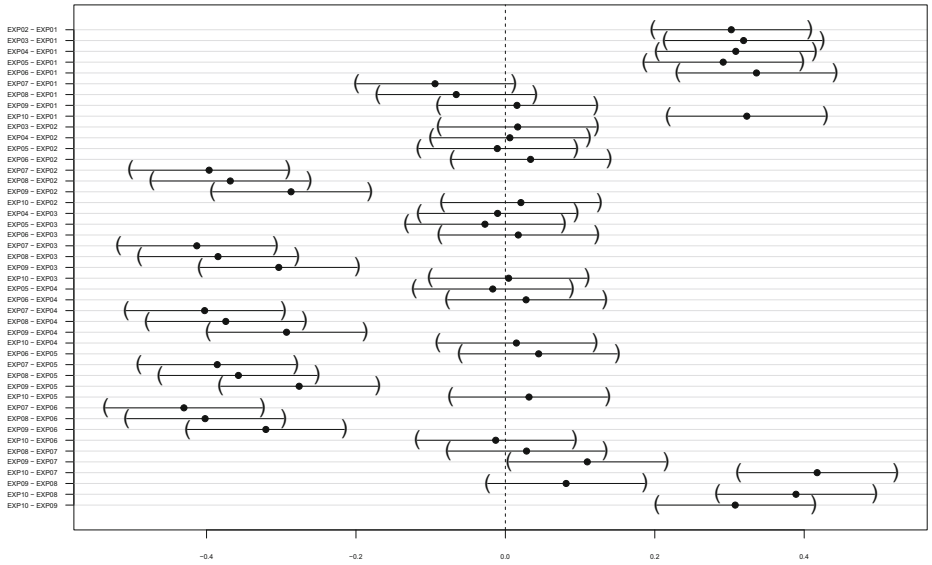


Fig. 21 Statistical comparison, all against all, of the 10 experiments with 95% confidence level

For all signs, the analysis of FP and FN confirms the result of the TP variable. Improving this accuracy may be possible by including depth image or performing an analysis based on phonological parameters such that CNN does not specialize in only sign location or movement.

6 Conclusion

Sign language differs from oral languages in its visual-gesture nature. It is composed of the configuration and movement of the hands, the orientation of the palm, the place where the sign is executed and the non-manual expressions. To identify a sign computationally is to interpret these phonological parameters. However, working with data in a logical time sequence demands high computational processing and data storage.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 9 Experiment 6: average performance metrics

Sign.	TP	FP	FN	Precision	Recall
To happen	1.00	0.00	0.10	1.00	0.91
Student	0.94	0.06	0.09	0.94	0.92
Yellow	0.98	0.02	0.05	0.98	0.95
America	0.96	0.04	0.02	0.96	0.98
To enjoy	0.97	0.04	0.09	0.96	0.92
Candy	0.90	0.10	0.03	0.90	0.97
Bank	0.93	0.07	0.20	0.93	0.82
Bathroom	0.97	0.04	0.03	0.96	0.97
Noise	0.93	0.07	0.00	0.93	1.00
Five	0.91	0.10	0.06	0.90	0.94
To know	1.00	0.00	0.21	1.00	0.83
Mirror	0.96	0.04	0.02	0.96	0.98
Corner	0.89	0.11	0.11	0.89	0.89
Son	0.97	0.03	0.01	0.97	0.99
Apple	0.98	0.02	0.13	0.98	0.88
Fear	0.84	0.17	0.05	0.83	0.94
Bad	0.92	0.08	0.01	0.92	0.99
Frog	0.88	0.13	0.08	0.87	0.92
Vaccine	0.90	0.11	0.05	0.89	0.95
Will	0.88	0.13	0.02	0.87	0.98
Average	0.93	0.07	0.07	0.93	0.94

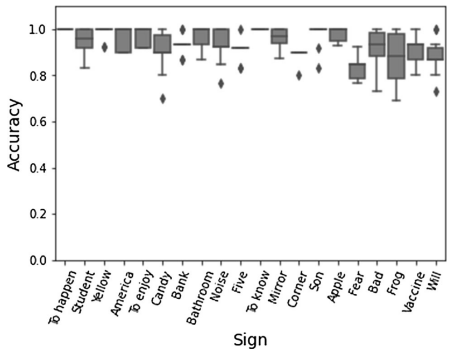


Fig. 22 Experiment 6: accuracy variance of each sign in 10 iterations

This article presents the creation of a Libras sign database, called MINDS-Libras, detailing the recording protocol used. This includes (1) a study of the basic requirements that a sign language database must have, (2) the signs chosen and the criteria for their selection, (3) how many signallers were chosen and what are their characteristics, (4) which sensors were used to capture the signs, (5) preparation of the recording environment, (6) standardization of the execution of each sign and, finally, (7) how and where these data will be made available. All these requirements were documented to make the MINDS-Libras replicable and reproducible.

MINDS-Libras has 20 signs recorded 5 times by 12 signallers. The database provides 1200 samples of the RGB-D sensor (Kinect v2 for Xbox One) recording and 995 of the RGB camera (Canon EOS Rebel t5). These data are videos in mp4 format of the RGB and depth data, ".txt" files of the body points and face information. It is available in [70].

Some approaches were implemented using RGB camera data and hand points provided by RGB-D sensor to classify these signs. Some empirical choices were made during the elaboration of the methodology and, in this article, it was possible to vary the network input, the data pre-processing and the type of CNN. The experiments showed that (1) the standardization of test data is important, (2) the technique of data augmentation contributed to improving the results and (3) the depth information (*z*-axis) should be considered in the recognition system. The best approach was using the grayscale image and data augmentation, obtaining 93.3% of accuracy on average.

For further development of the MINDS-Libras, future works are encouraged to follow the recording protocol described in this paper. However, any sensors and software that provide at least the videos in RGB can be used to generate the data, not being, therefore, mandatory to use the same setup as described here. Moreover, as hand movements are an important phonological parameter to the sign meaning, future works could further explore the hands points, focusing on the trajectory of the sign.

This work uses computational intelligence to minimize communication barriers and facilitate communication between those who have hearing impairments and those who do not. Libras currently is not a compulsory component of the basic school curriculum, which makes it difficult for deaf people to communicate with the majority of the population.

Acknowledgements The authors would like to thank Marcos Antônio Alves and Aline Xavier Fidêncio for the textual revision, and everyone who participated in the construction of the MINDS-Libras, especially all the people who volunteered their time to execute the signs. This work was partially financed by the Foundation for Research of the State of Minas Gerais [*Fundação de Amparo à*

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Pesquisa do Estado de Minas Gerais—FAPEMIG (Grant No. PPM-00587-16)], by the Coordination for the Improvement of Higher Education Personnel (*Coordenação de Aperfeiçoamento de Pessoal de Nível Superior*—CAPES), which is a Brazilian federal government agency under the Ministry of Education, by Federal Intitute of Minas Gerais (*Instituto Federal de Minas Gerais*—IFMG), and by the National Council for Scientific and Technological Development (*Conselho Nacional de Desenvolvimento Cientifico e Tecnológico*—CNPq), Brazil, Grants Nos. 306850/2016-8, 167016/2017-2 and 312991/2020-7, and Notice No. 169/2015.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Al-Hammadi M, Muhammad G, Abdul W, Alsulaiman M, Hossain MS (2019) Hand gesture recognition using 3D-CNN model. IEEE Consum Electron Mag 9(1):95–101
2. Al-Rousan M, Assaleh K, Tala'a A (2009) Video-based signer-independent Arabic sign language recognition using hidden Markov models. Appl Soft Comput 9(3):990–999
3. Almeida SGM (2014) Extração de características em reconhecimento de parâmetros fonológicos da Língua Brasileira de Sinais utilizando sensores RGB-D, Ph.D. thesis. Universidade Federal de Minas Gerais. https://www.ppgee.ufmg.br/defesas/303D.PDF. Accessed 03 Mar 2020. **(in Portuguese)**
4. Almeida SGM (2014) Libras-34 Dataset (Kinect v1). Zenodo. https://doi.org/10.5281/zenodo.4451526. Accessed 03 Sept 2020
5. Almeida SGM, Guimarães FG, Ramírez JA (2014) Feature extraction in Brazilian sign language recognition based on phonological structure and using RGB-D sensors. Expert Syst Appl 41(16):7259–7271
6. Almeida SGM, Guimarães FG, Ramírez JA (2015) Um método para sumarização de vídeos baseado no problema da diversidade máxima e em algoritmos evolucionários. In: XII Simpósio Brasileiro de Automação Inteligente (SBAI), Natal, Rio Grande do Norte, Brasil, pp 1298–1303 **(in Portuguese)**
7. Almeida SGM, Rezende TM, Toffolo ACR, Castro CL (2016) Libras-10 Dataset. https://doi.org/10.5281/zenodo.3229958
8. Aran O, Akarun L (2010) A multi-class classification strategy for Fisher scores: application to signer independent sign language recognition. Pattern Recogn 43(5):1776–1788
9. Aran O, Ari I, Akarun L, Sankur B, Benoit A, Caplier A, Campr P, Carrillo AH et al (2009) Signtutor: An interactive system for sign language tutoring. IEEE Multimed 81–93
10. Assaleh K (2005) Al-Rousan M (2005) Recognition of Arabic sign language alphabet using polynomial classifiers. EURASIP J Adv Signal Process 13:507614
11. Athira P, Sruthi C, Lijiya A (2019) A signer independent sign language recognition with co-articulation elimination from live videos: an Indian scenario. J King Saud Unive-Comput Inf Sci
12. Athitsos V, Neidle C, Sclaroff S (2008) American sign language Lexicon video dataset (ASLLVD). http://vlm1.uta.edu/∼athitsos/asl_lexicon/. Accessed 24 July 2020
13. Athitsos V, Neidle C, Sclaroff S, Nash J, Stefan A, Yuan Q, Thangali A (2008) The American sign language Lexicon video dataset. In: 2008 IEEE Computer Society conference on computer vision and pattern recognition workshops. IEEE, pp 1–8
14. Azar SG, Seyedarabi H (2020) Trajectory-based recognition of dynamic Persian sign language using hidden Markov model. Comput Speech Lang 61:101053
15. Beena M, Namboodiri A, Thottungal R (2020) Hybrid approaches of convolutional network and support vector machine for American sign language prediction. Multimed Tools Appl 79(5):4027–4040
16. Ben Tamou A, Ballihi L, Aboutajdine D (2017) Automatic learning of articulated skeletons based on mean of 3D joints for efficient action recognition. Int J Pattern Recogn Artif Intell 31(04):1750008
17. Bloom V, Argyriou V, Makris D (2016) Hierarchical transfer learning for online recognition of compound actions. Computer Vis Image Underst 144:62–72
18. Capovilla FC, Raphael WD, Temoteo JG, Martins AC (2017) Dicionário da Língua Brasileira do Brasil: A Libras em suas mãos, Volume I: Sinais de A a D., vol 1, 1st edn. Edusp, São Paulo **(in Portuguese)**
19. Capovilla FC, Raphael WD, Temoteo JG, Martins AC (2017) Dicionário da Língua Brasileira do Brasil: A Libras em suas mãos, Volume I: Sinais de A a D., vol 2, 1st edn. Edusp, São Paulo, Brasil **(in Portuguese)**
20. Capovilla FC, Raphael WD, Temoteo JG, Martins AC (2017) Dicionário da Língua Brasileira do Brasil: A Libras em suas mãos, Volume I: Sinais de P a Z., vol 3, 1st edn. Edusp, São Paulo, Brasil **(in Portuguese)**
21. Cardenas EE, Chavez GC (2020) Multimodal hand gesture recognition combining temporal and pose information based on CNN descriptors and histogram of cumulative magnitudes. J Vis Commun Image Represent 102772
22. Caselli NK, Sehyr ZS, Cohen-Goldberg AM, Emmorey K (2017) ASL-LEX: a lexical database of American sign language. http://asl-lex.org/. Accessed 13 May 2020
23. Castro GZ, Guerra RR, Assis MM, Rezende TM, Almeida GTB, Almeida SGM, Castro CL, Guimarães FG (2019) Desenvolvimento de uma base de dados de sinais de LIBRAS para aprendizado de máquina: estudo de caso com CNN 3D. In: 14ª Simpósio Brasileiro de Automação Inteligente, SBA, https://doi.org/10.17648/sbai-2019-111451 **(in Portuguese)**
24. Chadha A, Andreopoulos Y (2019) Improved techniques for adversarial discriminative domain adaptation. IEEE Trans Image Process 29:2622–2637
25. Chen FS, Fu CM, Huang CL (2003) Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vis Comput 21(8):745–758
26. Conly C, Doliotis P, Jangyodsuk P, Alonzo R, Athitsos V (2013) Toward a 3D body part detection video dataset and hand tracking benchmark. In: Proceedings of the 6th international conference on PErvasive technologies related to assistive environments. ACM, p 2
27. Cui R, Liu H, Zhang C (2019) A deep neural framework for continuous sign language recognition by iterative training. IEEE Trans Multimed 21(7):1880–1891
28. Cui Y, Weng J (2000) Appearance-based hand sign recognition from intensity image sequences. Comput Vis Image Underst 78(2):157–176
29. Dorner B, Hagen E (1994) Towards an American sign language interface. In: Integration of natural language and vision processing. Springer, Berlin, pp 143–161
30. Dreuw P, Rybach D, Deselaers T, Zahedi M, Ney H (2007) RWTH-BOSTON-104 Database. http://www-i6.informatik.rwth-aachen.de/∼dreuw/database-rwth-boston-104.php. Accessed 01 April 2020
31. Elakkiya R, Selvamani K (2017) Extricating manual and non-manual features for subunit level medical sign modelling in automatic sign language classification and recognition. J Med Syst 41(11):175
32. Elons AS, Abull-ela M, Tolba MF (2013) Neutralizing lighting non-homogeneity and background size in PCNN image signature for arabic sign language recognition. Neural Comput Appl 22(1):47–53
33. Escalera S, Gonzàlez J, Baró X, Reyes M, Lopes O, Guyon I, Athitsos V, Escalante H (2013) Multi-modal gesture recognition challenge 2013: dataset and results. In: Proceedings of the 15th ACM on international conference on multimodal interaction. ACM, pp 445–452
34. Escalera S, Athitsos V, Guyon I (2017) Challenges in multimodal gesture recognition. In: Gesture recognition. Springer, Berlin, pp 1–60
35. Escobedo-Cardenas E, Camara-Chavez G (2015) A robust gesture recognition using hand local data and skeleton trajectory. In: 2015 IEEE international conference on image processing (ICIP). IEEE, pp 1240–1244
36. Fagiani M, Principi E, Squartini S, Piazza F (2012) A new Italian sign language database. In: International conference on brain inspired cognitive systems. Springer, Berlin, pp 164–173
37. Fagiani M, Principi E, Squartini S, Piazza F (2015) Signer independent isolated Italian sign recognition based on hidden Markov models. Pattern Anal Appl 18(2):385–402
38. Felipe TA (2009) Libras em Contexto: Curso Básico: Livro do Estudante, 9th edn. WalPrint Gráfica Editora, Rio de Janeiro **(in Portuguese)**
39. Filho CFFC, de Souza RS, dos Santos JR, dos Santos BL, Costa MGF (2017) A fully automatic method for recognizing hand configurations of Brazilian sign language. Res Biomed Eng 33(1):78–89. https://doi.org/10.1590/2446-4740.03816
40. Forster J, Schmidt C, Hoyoux T, Koller O, Zelle U, Piater JH, Ney H (2012) RWTH-PHOENIX-Weather. http://www-i6.informatik.rwth-aachen.de/∼forster/database-rwth-phoenix.php. Accessed 13 May 2020
41. Freitas F, Barbosa F, Peres S (2014) Grammatical facial expressions data set. https://archive.ics.uci.edu/ml/datasets/Grammatical+Facial+Expressions. Accessed 16 Aug 2020
42. Freitas FA, Peres SM, Lima CAM, Barbosa FV (2014) Grammatical facial expressions recognition with machine learning. In: The Twenty-seventh international flairs conference
43. Guerra RR, Rezende TM, Guimaraes FG, Almeida SGM (2018) Facial expression analysis in brazilian sign language for sign recognition. In: Anais do XV Encontro Nacional de Inteligência Artificial e Computacional, SBC, pp 216–227 **(in Portuguese)**
44. Guo D, Zhou W, Li A, Li H, Wang M (2019) Hierarchical recurrent deep fusion using adaptive clip summarization for sign language translation. IEEE Trans Image Process 29:1575–1590
45. Hadfield S, Bowden R (2013) Scene particles: unregularized particle-based scene flow estimation. IEEE Trans Pattern Anal Machine Intell 36(3):564–576
46. Hasan MM, Misra PK (2011) Brightness factor matching for gesture recognition system using scaled normalization. Int J Comput Sci Inf Technol 3(2):35–46
47. Hassan M, Assaleh K, Shanableh T (2019) Multiple proposals for continuous Arabic sign language recognition. Sens Imaging 20(1):4
48. Hisham B, Hamouda A (2019) Supervised learning classifiers for Arabic gestures recognition using Kinect V2. SN Appl Sci 1(7):768
49. Holden EJ, Lee G, Owens R (2005) Australian sign language recognition. Mach Vis Appl 16(5):312
50. Honora M, Frizanco MLE (2010) Livro Ilustrado de Língua Brasileira de Sinais: Desvendando a Comunicação Usada Pelas Pessoas com Surdez. Ciranda Cultural, São Paulo **(in Portuguese)**
51. Ibrahim NB, Selim MM, Zayed HH (2018) An automatic Arabic sign language recognition system (ArSLRS). J King Saud Univ-Comput Inf Sci 30(4):470–477
52. Imran J, Raman B (2020) Deep motion templates and extreme learning machine for sign language recognition. Vis Comput 36(6):1233–1246
53. Infantino I, Rizzo R, Gaglio S (2007) A framework for sign language sentence recognition by commonsense context. IEEE Trans Syst Man Cybern Part C (Appl Rev) 37(5):1034–1039
54. Jadooki S, Mohamad D, Saba T, Almazyad AS, Rehman A (2017) Fused features mining for depth-based hand gesture recognition to classify blind human communication. Neural Comput Appl 28(11):3285–3294
55. Júnior PRM, De Souza RM, Werneck RdO, Stein BV, Pazinato DV, de Almeida WR, Penatti OA, Torres RdS, Rocha A (2017) Nearest neighbors distance ratio open-set classifier. Machine Learn 106(3):359–386
56. Kadous MW (2002) Australian sign language signs (high quality) Data Set. http://archive.ics.uci.edu/ml/datasets/Australian+Sign+Language+signs+(High+Quality). Accessed 19 July 2020
57. Kakoty NM, Sharma MD (2018) Recognition of sign language alphabets and numbers based on hand kinematics using A Data Glove. Proc Comput Sci 133:55–62
58. Kapuscinski T, Oszust M, Wysocki M, Warchol D (2015) Recognition of hand gestures observed by depth cameras. Int J Adv Robot Syst 12(4):36
59. Kawamoto A, Bertolini D, Barreto M (2018) A dataset for electromyography-based dactylology recognition. In: 2018 IEEE International conference on systems, man, and cybernetics (SMC). IEEE, pp 2376–2381
60. Kelly D, Mc Donald J, Markham C (2010) Weakly supervised training of a sign language recognition system using multiple instance learning density matrices. IEEE Trans Syst Man Cybern Part B (Cybern) 41(2):526–541
61. Koller O, Zargaran O, Ney H, Bowden R (2016) Deep sign: hybrid CNN-HMM for continuous sign language recognition. In: Proceedings of the British machine vision conference 2016
62. Kong W, Ranganath S (2014) Towards subject independent continuous sign language recognition: a segment and merge approach. Pattern Recogn 47(3):1294–1308
63. Kumar DA, Sastry A, Kishore P, Kumar EK (2018) 3D sign language recognition using spatio temporal graph kernels. J King Saud Univ-Comput Inf Sci
64. Li W (2017) Webpage of Dr Wanqing Li. http://www.uow.edu.au/∼wanqing/#MSRAction3DDatasets. Accessed 10 July 2020
65. Li W, Zhang Z, Liu Z (2010) Action recognition based on a bag of 3D points. In: 2010 IEEE Computer Society conference on computer vision and pattern recognition-workshops. IEEE, pp 9–14
66. Liao Y, Xiong P, Min W, Min W, Lu J (2019) Dynamic sign language recognition based on video sequence with BLSTM-3D residual networks. IEEE Access 7:38044–38054
67. Lim KM, Tan AW, Tan SC (2016) A feature covariance matrix with serial particle filter for isolated sign language recognition. Expert Syst Appl 54:208–218
68. Liu L, Shao L (2013) Learning discriminative representations from RGB-D video data. In: Twenty-third international joint conference on artificial intelligence
69. Lucas BD, Kanade T, et al. (1981) An iterative image registration technique with an application to stereo vision. In: Proceedings DARPA image understanding workshop
70. Machine Intelligence and Data Science Laboratory (Minds Lab) (2019) Brazilian sign language recognition. http://minds.eng.ufmg.br/project/4. Accessed 04 Sept 2020

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Neural Computing and Applications

Springer

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

71. Machine Vision Lab (2018) IITR Sign Language Thermal Dataset 2018 (ISLTD2018). https://www.iitr.ac.in/mvlab/documents/ISLTD2018_Download_Form.pdf. Accessed 03 Sept 2020
72. Masood S, Srivastava A, Thuwal HC, Ahmad M (2018) Real-time sign language gesture (word) recognition from video sequences using CNN and RNN. In: Intelligent engineering informatics. Springer, pp 623–632
73. MCC Lab (2020) SLR Dataset. http://mccipc.ustc.edu.cn/mediawiki/index.php/SLR_Dataset. Accessed 03 Sept 2020
74. Mohandes M, Deriche M, Johar U, Ilyas S (2012) A signer-independent Arabic Sign Language recognition system using face detection, geometric features, and a hidden Markov model. Comput Electr Eng 38(2):422–433
75. Mohandes MA (2013) Recognition of two-handed Arabic signs using the CyberGlove. Arabian J Sci Eng 38(3):669–677
76. Molchanov P, Gupta S, Kim K, Kautz J (2015) Hand gesture recognition with 3D convolutional neural networks. Proceedings of the IEEE conference on computer vision and pattern recognition workshops, vol 1. IEEE, pp 1–7
77. Nguyen TD, Ranganath S (2012) Facial expressions in American sign language: tracking and recognition. Pattern Recogn 45(5):1877–1891
78. Oszust M, Wysocki M (2013) Polish sign language words recognition with kinect. In: 2013 6th international conference on human system Interactions (HSI). IEEE, pp 219–226
79. Oszust M, Wysocki M (2016) Point clouds corresponding to dynamic gestures registered by Kinect. http://vision.kia.prz.edu.pl/dynamickinect.php. Accessed 13 May 2020
80. Oszust M, Wysocki M (2016) Point clouds corresponding to dynamic gestures registered by time-of-flight (ToF) camera. http://vision.kia.prz.edu.pl/dynamictof.php. Accessed 15 May 2020
81. Oz C, Leu MC (2011) American Sign Language word recognition with a sensory glove using artificial neural networks. Eng Appl Artif Intell 24(7):1204–1213
82. Ozcan T, Basturk A (2019) Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition. Neural Comput Appl 31(12):8955–8970
83. Raghuveera T, Deepthi R, Mangalashri R, Akshaya R (2020) A depth-based Indian Sign Language recognition using Microsoft Kinect. Sādhanā 45(1):34
84. Rastgoo R, Kiani K, Escalera S (2020) Hand sign language recognition using multi-view hand skeleton. Expert Syst Appl 150:113336
85. Ravi S, Suman M, Kishore P, Kumar K, Kumar A et al (2019) Multi modal spatio temporal co-trained CNNs with single modal testing on RGB-D based sign language gesture recognition. J Comput Lang 52:88–102
86. Rezende TM (2016) Aplicação de Técnicas de Inteligência Computacional para Análise da Expressão Facial em Reconhecimento de Sinais de Libras. Master's thesis, Universidade Federal de Minas Gerais. https://www.ppgee.ufmg.br/defesas/1393M.PDF. Accessed 03 Sept 2020. **(in Portuguese)**
87. Rezende TM, de Castro CL, Moreira SG, Preto CO (2017) Análise da expressao facial em reconhecimento de sinais de libras. In: VI Simpósio Brasileiro de Automação inteligente, pp 465–470 **(in Portuguese)**
88. Ronchetti F, Quiroga F, Estrebou C, Lanzarini L, Rosete A (2016) LSA64: a dataset for Argentinian sign language. http://facundoq.github.io/unlp/lsa64/index.html. Accessed 03 Aug 2020
89. Ruffieux S, Lalanne D, Mugellini E, Khaled OA (2014) A survey of datasets for human gesture recognition. In: International conference on human–computer interaction. Springer, Berlin, pp 337–348
90. Shi J et al (1994) Good features to track. In: 1994 Proceedings of IEEE conference on computer vision and pattern recognition. IEEE, pp 593–600
91. Simons GF, Fennig CD (2018) Ethnologue: languages of the World. SIL International, Dalas, Texas. https://www.ethnologue.com/subgroups/sign-language. Accessed 18 Aug 2020
92. Stokoe WC (1960) Sign language structure: an outline of the visual communication systems of American deaf. University of Buffalo Press, New York
93. Tamura S, Kawasaki S (1988) Recognition of sign language motion images. Pattern Recogn 21(4):343–353. https://doi.org/10.1016/0031-3203(88)90048-9
94. Terven JR, Córdova-Esparza DM (2016) Kin2. A Kinect 2 toolbox for MATLAB. Sci Comput Program 130:97–106
95. Tolba MF, Samir A, Aboul-Ela M (2013) Arabic sign language continuous sentences recognition using PCNN and graph matching. Neural Comput Appl 23(3–4):999–1010
96. Tran D, Bourdev LD, Fergus R, Torresani L, Paluri M (2014) C3D: Generic Features for Video Analysis. CoRR. arXiv:arXiv:1412.0767
97. Tubaiz N, Shanableh T, Assaleh K (2015) Glove-based continuous Arabic sign language recognition in user-dependent mode. IEEE Trans Hum-Mach Syst 45(4):526–533
98. Vogler C, Goldenstein S (2008) Facial movement analysis in ASL. Univers Access Inf Soc 6(4):363–374
99. Von Agris U (2008) Database for signer-independent continuous sign language recognition. https://www.phonetik.uni-muenchen.de/forschung/Bas/SIGNUM/. Accessed 13 May 2020
100. Wadhawan A, Kumar P (2020) Deep learning-based sign language recognition system for static signs. Neural Comput Appl 32(12):7957–7968
101. Wang H, Chai X, Hong X, Zhao G, Chen X (2016) Isolated sign language recognition with grassmann covariance matrices. ACM Trans Access Comput (TACCESS) 8(4):1–21
102. Wang H, Chai X, Chen X (2019) A novel sign language recognition framework using hierarchical Grassmann covariance matrix. IEEE Trans Multimed 21(11):2806–2814
103. Xia L, Chen CC, Aggarwal JK (2011) Human detection using depth information by kinect. In: CVPR 2011 workshops. IEEE, pp 15–22
104. Zahedi M, Keysers D, Deselaers T, Ney H (2005) RWTH-BOSTON-50 Database. https://www-i6.informatik.rwth-aachen.de/web/Software/Databases/Signlanguage/details/rwth-boston-50/index.php. Accessed 13 May 2020
105. Zhang L, Zhu G, Shen P, Song J, Afaq Shah S, Bennamoun M (2017) Learning spatiotemporal features using 3DCNN and convolutional LSTM for gesture recognition. In: Proceedings of the IEEE international conference on computer vision, pp 3120–3128
106. Zhao R, Martinez AM (2015) Labeled graph kernel for behavior analysis. IEEE Trans Pattern Anal Mach Intell 38(8):1640–1650

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH ("Springer Nature").

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users ("Users"), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use ("Terms"). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com

C.4 Trabalhos complementares.

1. **Artigo**: Otimizaçao Dinâmica Evolucionária para Despacho de Energia em uma Microrrede usando Veıculos Elétricos.

   Evento: 14º Simpósio Brasileiro de Automação Inteligente (SBAI 2019).

   Data: 27 a 30 de outubro de 2019.

   Local: Ouro Preto - Minas Gerais.

   DOI: 10.17648/sbai-2019-111524

2. ***Live***: É possível reconhecer sinais de Libras com Machine Learning?

   Evento: AI Girls Comunidade.

   Data: 12 de agosto de 2020.

   Link: https://www.youtube.com/watch?v=Hzg_Y_fza6M&t=149s.

3. ***Live***: É possível reconhecer sinais de Libras com Machine Learning?

   Evento: DWC - Trilha: Ciência de Dados, Machine Learning e Inteligência Artificial - DevelopersBr.

   Data: 26 de setembro de 2020.

   Link: https://www.youtube.com/watch?v=fRbPg_8HWZg&t=3798s.

4. **Palestra e Mesa-Redonda**: Abrindo a Caixa Preta da IA

   Evento: XIII Semana de Ciência e Tecnologia 2020 - Instituto Federal de Minas Gerais - *Campus* Ouro Preto.

   Data: 19 de outubro de 2020.

   Link: https://www.youtube.com/watch?v=88xunY3xCTw&t=521s.

5. **Artigo**: Metodologia ativa no ensino técnico: ensinando conceitos básicos de Instrumentação e Controle de Processos.

   Evento: XXIII Congresso Brasileiro de Automática (CBA 2020).

   Data: 23 a 26 de novembro de 2020.

   DOI: 10.48011/asba.v2i1.1710

6. **Artigo**: Aprendizado Ativo via Algoritmo de Evolução Diferencial.

   Evento: XXIII Congresso Brasileiro de Automática (CBA 2020).

   Data: 23 a 26 de novembro de 2020.

   DOI: 10.48011/asba.v2i1.1726

7. ***Live***: Como encontrar a melhor técnica para os meus dados? Existe um passo-a-passo?

   Evento: Aniversário de 1 ano da Comunidade AI Girls - Trilha: Inteligência Artificial.

   Data: 27 de fevereiro de 2021.

   Link: https://www.youtube.com/watch?v=bPDuZcpgeyA.

8. ***Live***: Inclusão das mulheres na Inteligência Artificial

   Evento: Maratona de Estatística e Ciência de Dados / Comunidade EstaTiDados.

   Data: 10 de março de 2021.

   Link: https://www.youtube.com/watch?v=SjJ3zYMY79g&t=2834s.

9. **Capítulo de Livro**: Metodologia ativa no ensino técnico: ensinando conceitos básicos de Instrumentação e Controle de Processos.

   Livro: Educação Contemporânea – Volume 23.

   Editora: Poisson.

   Março de 2021.

   DOI: 10.36229/978-65-5866-098-9.CAP.13

10. **Entrevista**: Mulheres na tecnologia

    Revista: *Female Tech Leaders Magazine*.

    Data: Abril de 2021.

    Link: https://bit.ly/3nByZmM.

11. **E-book**: Editor de texto e imagem: Uso da ferramenta LaTeX

    Editor: Formação Inicial e Continuada - Instituto Federal de Minas Gerais.

    Data: Maio de 2021.

    ISBN: 978-65-5876-109-9.

    Link: https://mais.ifmg.edu.br/course/index.php?categoryid=12.

12. **Palestra**: Aplicações de Visão Computacional: Reconhecimento de Sinais de Libras e Detecção de Incêndio.

    Evento: Papo de Engenharia - Automic Jr. - Engenharia de Controle e Automação / UFOP.

    Data: 25 de maio de 2021.

    Meio de divulgação: *Google Meeting*.

13. **Palestra**: Programas de pós-graduação, Inteligência Artificial e Reconhecimento Automático de Sinais de Libras.

    Evento: I Semana Crea-Jr / UFOP.

    Data: 28 de maio de 2021.

    Meio de divulgação: *Zoom*.

14. **Palestra**: Reconhecimento Automático de Sinais de Libras: uma aplicação de Inteligência Computacional.

    Evento: V Semana da Engenharia / Instituto Federal de Minas Gerais - *Campus* Avançado Itabirito.

    Data: 07 de julho de 2021.

    Meio de divulgação: https://www.youtube.com/watch?v=CdbpPh2Ev7w&t=55s&ab_channel=IFMGItabirito.

15. **Artigo**: Fire Detection based on a Two-Dimensional Convolutional Neural Network and Temporal Analysis.

    Evento: 7th Latin American Conference on Computational Intelligence (IEEE LA-CCI).

    Data: 2 a 4 de novembro de 2021.

    Situação: aceito.

16. **Resumo**: Sistema de detecção de incêndios utilizando inteligência artificial com participação da sociedade: Projeto ApagaOFogo.

    Evento: XXVI Seminário Nacional de Produção e Transmissão de Energia Elétrica (SNPTEE).

    Data: 15 a 18 de maio de 2022.

    Situação: submetido.