UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

MATHEUS MACÊDO CLAUDINO

RECONHECIMENTO DE LIBRAS EM FRAMES ESTÁTICOS DE VÍDEOS UTILIZANDO CNN E TÉCNICAS DE PRÉ-PROCESSAMENTO DE IMAGENS

CAMPINA GRANDE - PB

2022

MATHEUS MACÊDO CLAUDINO

RECONHECIMENTO DE LIBRAS EM FRAMES ESTÁTICOS DE VÍDEOS UTILIZANDO CNN E TÉCNICAS DE PRÉ-PROCESSAMENTO DE IMAGENS

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador : Herman Martins Gomes

CAMPINA GRANDE - PB

2022

MATHEUS MACÊDO CLAUDINO

RECONHECIMENTO DE LIBRAS EM FRAMES ESTÁTICOS DE VÍDEOS UTILIZANDO CNN E TÉCNICAS DE PRÉ-PROCESSAMENTO DE IMAGENS

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Herman Martins Gomes

Orientador – UASC/CEEI/UFCG

Hyggo Oliveira De Almeida

Examinador – UASC/CEEI/UFCG

Francisco Vilar Brasileiro

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 02 de Setembro de 2022.

CAMPINA GRANDE - PB

RESUMO (ABSTRACT)

Although Brazilian sign language (Libras) was recognized as an official language of Brazil in 2002 [1], legal measures that regulate and require the offer of Libras teaching in schools to some degree were only reversed in a bill in 2019 [2]. Resulting in a lack of contact of people without hearing problems with Libras, and combined with the finding of the World Federation of the Deaf (WFD) that about 80% of the deaf in the world have problems understanding the written languages of their respective countries [3], generates social isolation with people who depends on the use of hand signals to communicate. In this context, there is the field of recognition of signal languages (RLS), which proposes to create technological interfaces that can act on the described problem. This work uses static video frames extracted from the MINDS-Libras dataset [17] to analyze the impact of using image preprocessing methods in the training of a Convolutional Neural Network (CNN), to obtain a model capable of classifying 20 different signals of Pounds efficiently. In the end, the proposed method reached an average accuracy of 91.08% in the data set used.

Reconhecimento de Libras em frames estáticos de vídeos utilizando CNN e técnicas de pré-processamento de imagens

Matheus Macêdo Claudino
Graduando em Ciência da Computação Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
matheus.claudino@ccc.ufcg.edu.br

Herman Martins Gomes
Unidade Acadêmica de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
hmg@computacao.ufcg.edu.br

RESUMO

Embora a língua de sinais brasileira (Libras) tenha sido reconhecida como uma língua oficial do Brasil em 2002 [1], medidas legais que regularizem e exijam a oferta de ensino de Libras nas escolas em algum grau, apenas foram revertidas em um projeto de lei em 2019 [2]. Resultando numa falta de contato de pessoas sem problemas auditivos com Libras, e combinado à constatação da World Federation of the Deaf (WFD) de que cerca de 80% dos surdos do mundo possuem problemas de compreensão nas línguas escritas de seus respectivos países [3], gera um isolamento social daqueles que dependem do uso de Libras para se comunicar. Neste contexto, existe o campo de reconhecimento de linguagens de sinais (RLS), que se propõe a criar interfaces tecnológicas que possam atuar no problema descrito. Este trabalho utiliza de quadros de vídeos estáticos extraídos do dataset MINDS-Libras [17] para analisar o impacto do uso de métodos de pré-processamento de imagens no treinamento de uma Rede Neural Convolucional (CNN), com o intuito de se obter um modelo capaz de classificar 20 sinais diversos de Libras de forma eficiente. Ao final, o método proposto alcançou acurácia média de 91.08% no conjunto de dados utilizado.

Palavras-chave

CNN, Libras, Pré-processamento de imagens, MINDS-Libras.

1. INTRODUÇÃO

Libras é um idioma utilizado majoritariamente pela comunidade surda brasileira, que usa de movimentos de mãos, posição corporal e expressões faciais para realizar a comunicação [4]. Através dela, assim como em qualquer outra língua, é possível realizar a expressão de conceitos, ideias, sentimentos, pensamentos ou opiniões. Sua estrutura gramatical é independente da Língua portuguesa falada no Brasil [4], ou seja, características da língua brasileira falada e escrita como: pronomes, sujeito e predicado, não estão necessariamente presentes nessa gramática.

De acordo com estudos realizados pela World Federation of the Deaf (WFD), em 2004, 80% dos surdos do mundo apresentavam baixa escolaridade e apresentavam problemas de compreensão das línguas escritas de seus países, em ambos campos de leitura e escrita [3]. Isto gera um problema social grave pois, embora a lei brasileira garanta aos deficientes auditivos direito de igualdade [5], na prática, os surdos que não dominam a língua escrita não conseguem se comunicar plenamente com o restante da sociedade. E mesmo que deficientes auditivos tentem fazer uso de Libras para se comunicarem, esta linguagem não é difundida no país, como constatado pelo IBGE na Pesquisa Nacional de Saúde de 2019, onde foi apurado que apenas cerca de 22,4% das pessoas de 5 a 40 anos com algum grau de deficiência auditiva sabiam usar a Libras [6].

Essa barreira de comunicação gera um isolamento social entre aqueles que dependem do uso de Libras para se comunicar e o resto da sociedade, podendo gerar impactos psicológicos graves em situações onde, por exemplo, famílias de crianças surdas não possuem acesso às informações necessárias para garantir uma criação sem a presença de estigmas depreciativos [3]. A situação se agrava ainda mais quando em contextos onde o sucesso da comunicação é crítico para que não aconteçam problemas graves para o deficiente auditivo, tal como a relação médico paciente na utilização de serviços de saúde [9].

Com o objetivo de reduzir essa barreira de linguagem, esforços vêm sendo empenhados no campo de reconhecimento de linguagens de sinais (RLS) [9-10]. Uma das formas que esse campo pode ajudar nessa interação é através de sistemas automáticos de tradução de linguagem de sinais para linguagem textual, entretanto o processo de desenvolvimento desses tradutores não é direto ao ponto, pois existe uma variabilidade de cenários, desde a variabilidade de movimentos de cada língua, bem como como possíveis vícios que o locutor possa ter ao performar os movimentos.

Nas línguas de sinais, os campos visuais e espaciais são imprescindíveis, já que os sinais são produzidos combinando postura corporal, expressão facial, movimento de mãos e formato dos símbolos [8,12]. Assim, técnicas relacionadas à visão computacional estão entre os estudos que ganharam relevância nos últimos anos, apresentando resultados interessantes sem a necessidade de dispositivos dedicados como braceletes ou luvas, assim sendo uma abordagem mais barata [8].

Existe uma forte variância linguística entre as línguas de sinais e as línguas faladas [9], dado que é comum que ambas tenham sua estrutura sintática e gramatical estruturadas de formas diferente, por exemplo, a sintaxe da língua de sinais americana ASL é mais parecida com o Japonês do que com o Inglês [10]. Ao considerar este processo de adequação gramatical e sintática entre linguagens como um problema à parte, o campo de RLS aborda o problema de realizar interpretações das linguagem de sinais em

texto, como sendo meramente uma tarefa de reconhecimento de visão computacional. Ignorando assim a profundidade da lógica gramatical e estrutura linguística presente nos sinais.

2. FUNDAMENTAÇÃO TEÓRICA

Quando se trata da escolha de abordagem a ser utilizada no campo de visão computacional relacionada a RLS, é importante considerar o tipo de dado que será trabalhado e o meio com qual o dado será captado. Como um todo, as abordagens dos trabalhos desta área podem ser divididas em dois grupos: aqueles que fazem uso de imagens e vídeos capturados por câmeras convencionais e aqueles que usam dados capturados por dispositivos dedicados. A câmera de profundidade RGB-D é um exemplo deste tipo de dispositivo [8,13], seu uso tem sido um marco importante na comunidade de visão computacional, pois provém outras dimensões de dados além da imagem capturada, tais como profundidade de imagem, esqueleto corporal autogerado e silhuetas de usuários.

Um ponto importante no desenvolvimento de RLS é a definição do escopo de dados trabalhados, pois existem duas vertentes quando se trata do reconhecimento de sinais: o reconhecimento contínuo e o estático. No primeiro não se possui informações dos quadros que delimitam a ocorrência de um sinal em determinado vídeo, sendo essa delimitação entre o começo e o fim da execução do gesto, sendo identificado uma das tarefas que o modelo gerado terá que executar. Já o reconhecimento estático por sua vez possui essas informações durante a execução do reconhecimento.

Stefano *et al.* [11] propôs um método semi-supervisionado para identificar e classificar sinais de Libras a partir de vídeos, utilizando um conjunto de dados mais próximo de cenários reais, com grande variação de intérprete, pose e cenário. O trabalho realiza a coleta dos dados de treinamento em vídeos no Youtube, os identificando pela variação de intensidade de movimento e catalogando os dados de acordo com suas legendas, sendo obtido assim um dataset de vídeos com mais de 110 sinais diferentes, nos quais 105 são dinâmicos e 5 são estáticos. Cada frame dos vídeos foi submetido a uma conversão em níveis de cinza e binarizados utilizando o método de Otsu [18] adaptativo. Para realização da extração de características é utilizado uma Convolutional Neural Network (CNN) que realiza a análise de 18 frames extraídos uniformemente de cada vídeo e aplica uma Long Short-Term Memory (LTSM) para realizar a correlação temporal entre as 18 características geradas, para assim classificar o sinal. O método obteve acurácia de 61.6%, se destacando por sua variabilidade de dados e por comparar aplicar o método nas arquiteturas VGG16, VGG19 e InceptionV3.

Vidalón *et al*. [8] propôs um método de classificação de 107 sinais de Libras de cunho médico a partir de dados capturados por um Kinect [19]. Para segmentar os sinais do fluxo contínuo, o sistema detecta padrões de início e pausa de um movimentos baseado na informação de esqueleto gerada pelo Kinect, e após segmentado o sinal é utilizado um classificador Dynamic Time Warping–Nearest Neighbor (DTW-kNN) para identificar o sinal. Os dados de treinamento foram coletados em um ambiente controlado com distância fixa entre sinalizador e câmera. Este trabalho se destaca por alcançar uma acurácia de 98.69%, entretanto falha em identificar sinais contínuos, visto que a identificação do sinal é feita a partir da análise de movimento.

Yin [12] propôs um método para tradução de vídeos utilizando dois datasets de palavras e frases de reconhecimento contínuo, o RWTH-PHOENIX-Weather 2014T composto por vídeos da linguagem de sinais alemã coletados de programas televisivos de previsão de tempo, e o ASLG-PC12 composto por traduções de sinais da linguagem de sinais americana para texto. Não foi aplicado nenhuma técnica de pré-processamento de imagem para os frames dos vídeos utilizados no desenvolvimento do modelo. Para realizar a tradução dos vídeos foi utilizado uma arquitetura de redes neurais Transformer com para extração de significados literais dos sinais e outra rede de arquitetura Spatio-Temporal Multi-Cue (STMC) para realizar a tradução de um conjunto de significado literal para texto. Este trabalho se destaca por apresentar uma abordagem diferente das comumente encontradas para extração de características, conseguindo alcançar resultados promissores ao elevar cerca de 5 e 7 pontos o valor de pontuação BLEU-4 para o dataset de vídeos RWTH-PHOENIX-Weather 2014T.

Bheda el al [13]. propôs um método de classificação de imagens focalizadas em uma única mão, a qual representa uma das 26 letras do alfabeto da linguagem de sinais americana através de uma arquitetura de CNN. Os dados foram criados pelos criadores deste trabalho, o qual foram submetidos ao pré-processamento de extração de background e aumento de dados por rotação e inversão vertical de imagem. Este trabalho ao final obteve uma acurácia de 82.5% para os dados de treinamento e 97% para os dados de validação, se destacando por mostrar que o uso de camadas de dropout em CNN ajuda a melhorar a acurácia no conjunto de validação e mostra efetividade do uso de técnicas de aumento de dados para este tipo de problema, entretanto vale se elencar dúvidas quanto a confiabilidade dos resultados, visto que foi utilizado um dataset autogerado em condições de laboratório.

Lazo *et al.* [14] apresenta um trabalho propôs um método semelhante à [13] para reconhecer o conjunto de letras do alfabeto da linguagem de sinais peruana, utilizando um conjunto de dados autogerado com a adição de um segmentador de mãos para utilizar uma imagem binarizada de apenas a mão do sinalizador para realizar a predição. Ao final, a abordagem utilizada obteve uma acurácia de 99.85% que embora relevante, ainda é limitada pois apenas pode ser utilizada para sinais que utilizem apenas o formato das mãos e não possuam variações de ângulo bruscas.

Sharma *et al.* [15] propôs um método de classificação de imagens de mãos em letras do alfabeto da linguagem americana de sinais. É utilizado um dataset próprio e pouco diverso, o qual é submetido a um pré-processamento de imagem com transformação em níveis de cinza, seguido pelo uso detector de bordas Canny. E para extração de feature é utilizado a técnica de Oriented Fast and Rotated Brief (ORB), sendo os resultados submetidos a um conjunto de classificadores: Random Forests, Support Vector Machines, Naïve Bayes, Logistic Regression, K-Nearest Neighbours e Multilayer Perceptron. Este trabalho se destaca pelo uso de ORB como técnica de extração de feature em conjunto a um pré-processamento de imagens utilizadas obtendo altas acurácias com os classificadores, embora use um dataset pouco representativo.

Barros *et al.* [16] desenvolveu uma arquitetura CNN para análise de posição e formato de mãos com três canais de informação, onde um leva uma imagem em níveis de cinza e o segundo e terceiro recebem mapas de bordas resultantes da aplicação do filtro Sobel horizontal e vertical. Embora este trabalho não envolva linguagens de sinais diretamente, ele obtém ótimos resultados num dataset representativo, com taxas de acurácia e F1-score em torno dos 90%.

Rezende *et al.* [17] desenvolveu um conjunto de dados composto por 20 sinais repetidos 5 vezes por 12 intérpretes, totalizando 1.200 sequências de dados de vídeo gravados simultaneamente por um Microsoft Kinect v2 e uma DSLR da Canon, sendo esse conjunto de dados conhecido como MINDS-Libras. E em conjunto ao conjunto de dados foi desenvolvido um sistema de reconhecimento empregando Redes Convolucionais tridimensionais (3D-CNN), sendo testado diferentes abordagens de extração de características nas imagens em nível de cinza. O melhor resultado obtido teve uma acurácia média de 93.3%, se destacando por apresentar uma arquitetura em nível de estado da arte para classificação de vídeos e o desenvolvimento de um conjunto de dados grande e bem diverso quando comparado com similares.

Ao observar os trabalhos citados acima, notou-se que mesmo apresentando abordagens diversificadas, é possível elencar uma característica comum entre eles: pouca ou nenhuma preocupação é empenhada para comparar o impacto de diferentes técnicas de pré-processamento de imagens no resultado de classificação final. Onde a transformação da imagem para níveis de cinza é a abordagem mais comum, com um ou outro apresentando uma variação usando detectores de bordas Canny ou Sobel, subtração de background ou uso de mapas de saliência.

3. DESCRIÇÃO DA SOLUÇÃO

Com o intuito de se desenvolver um modelo de aprendizagem de máquina capaz de reconhecer um conjunto limitado e pré-determinado de símbolos de Libras, utilizando quadros estáticos de vídeos, foi escolhida a abordagem de CNN-2D, pois é uma abordagem que vem se destacando na área de RLS, junto à CNN-3D, por serem efetivas na representação de modelações espaço-temporais de vídeos e imagens de linguagens de sinais. E para verificar possíveis efeitos do uso de diferentes técnicas de processamento de imagens nos quadros utilizados no treinamento dos modelos, foi instaurada uma arquitetura padrão de CNN-2D a ser utilizada nos treinamentos com o mesmo número de épocas e hiperparâmetros, modificando apenas as técnicas de pré-processamento utilizadas.

3.1 Ambiente de desenvolvimento

Na realização das atividades e processos descritos nesta seção foi utilizado a plataforma Google Collaboratory com uso de GPU para realizar a execução de todos os scripts de manipulação de imagens e de treinamento. Para o desenvolvimento da arquitetura e treinamentos de modelos CNN, foi utilizado as bibliotecas Keras e Tensor Flow; e em conjunto foi utilizado a biblioteca OpenCV para realizar manipulações de vídeos e imagens necessárias.

3.2 Arquitetura

Foi escolhida uma arquitetura básica de CNN na qual foram utilizadas 3 sequências de convolução 2D com Relu alternadas com camadas de Max Pooling, seguidas por dois grupos de camada totalmente conectada seguidas por uma camada de dropout após a camada Flatten. Para o treinamento da CNN foi utilizado o otimizador Adam com learning rate de 0.001 em conjunto com a função de cálculo de Loss: Sparse Categorical Cross Entropy. A estrutura detalhada da CNN utilizada pode ser vista na Tabela 1. Vale destacar que existem duas variações desta arquitetura, uma com camada de entrada de dados com shape 200 x 180 x 3, para quando se usa imagens coloridas e uma com shape 200 x 180 x 1, para quando se usa imagens em níveis de cinza.

Tabela 1: Arquitetura de CNN utilizada.

Camada	Tipo de camada	Formato saída
input	entrada de dados	200 x 180 x 3
conv1	convolução 2D + ReLU	200 x 180 x 32
pool1	max-pooling	100 x 90 x 32
conv2	convolução 2D + ReLU	100 x 90 x 32
pool2	max-pooling	50 x 45 x 64
conv3	convolução 2D + ReLU	50 x 45 x 64
pool3	max-pooling	25 x 22 x 64
flatten	Flatten	35200
dropout1	Dropout 20%	35200
fc1	Totalmente conectada + ReLU	64
dropout2	Dropout 20%	64
fc2	Totalmente conectada	20

Fonte: Autoria própria.

3.3 Dados

A origem dos dados utilizados no treinamento são vídeos do banco de dados conhecido por: MNDIS-Libras. Desenvolvido por Rezende *et al*. [17], o banco de dados é composto por 20 sinais de libras, indexados com um número de 1 à 20, efetuados um total de 5 vezes em diferentes condições de iluminação e posição corporal por 12 sinalizadores com diferentes níveis de proficiência e proximidade com Libras. O conjunto de sinalizadores embora não seja perfeitamente balanceado é diverso, tendo um total de 4 sinalizadores masculinos e 8 sinalizadores femininos, possuindo grande variabilidade étnica e corporal para ambos os gêneros. Entretanto, ele continua sendo um conjunto de laboratório, logo os resultados obtidos para esse conjunto pré-determinado de sinais, pode não ser válido para ambientes com alto grau de interferências, tais como iluminação e posição de câmera.

Nas performances dos sinalizadores é possível visualizar a parte superior do corpo do corpo, a partir da cintura com o sinalizador, paralelo a câmera com um fundo composto por uma tela verde. Os dados foram capturados por um Kinect, portanto possui dados de profundidade, formato corporal e RGB

associado a cada execução de sinal, destes o único dado utilizado foi os vídeos RGB no formato MP4 na resolução de 1920 x 1080.

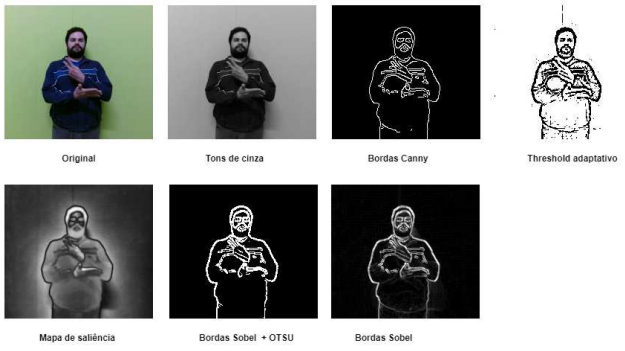
Um dos desafios encontrados no uso deste banco de dados, foi o método de extração de frames estáticos dos vídeos, pois todos os 20 sinais representam sinais compostos com múltiplas etapas de performance. Para contornar isto, para cada uma das classes, foram escolhidos 3 frames por vídeo, onde o conjunto final de frames extraídos para uma mesma classe devem representar a mesma etapa de performance do sinalizador, com mesma configuração de mãos, posição corporal e expressão facial. Ao final, se obteve um dataset de 3600 imagens estáticas usadas no desenvolvimento dos modelos, com uma divisão de 80% dos dados a serem usados no treinamento e 20% a serem usados na validação.

3.4 Pré-processamento de imagens

Com o intuito de reduzir a quantidade de informações a serem utilizadas na rede neural foi feita uma análise manual para checar quais pixels do conjunto de frames estáticos do banco de dados possuem informações do sinal em questão, onde se percebeu que existiam bordas desnecessárias sem nenhuma informação relevante. Portanto foi feito o corte dessas bordas, se obtendo vídeos na resolução de 1200 x 1080, os quais foram redimensionados para um sexto do seu tamanho original no formato de 200 x 180. Vale destacar que antes de serem treinados, todos os os valores presentes nos canais de imagem dos dados de treinamento e validação foram normalizados no intervalo de 0 a 1.

Quanto às transformações de imagem testadas com a arquitetura proposta, foram efetuadas diferentes combinações entre as encontradas na pesquisa de trabalhos relacionados testando a efetividade das transformações: para tons de cinza, detectores de borda, binarização de imagens (Threshold) e mapas de saliência. Na Figura 1 é possível comparar a imagem original com as 6 técnicas de transformações de imagem propostas, onde para cada uma das técnicas, foi treinado e avaliado um modelo. Destaca-se que antes da aplicação de cada uma das técnicas de pré-processamento de imagem, foi aplicada a conversão da imagem para tons de cinza e aplicado um filtro de desfoque Gaussiano.

Figura 1: Transformações de imagem testadas.



Fonte: Autoria própria.

3.5 Aumento de dados

Antes da realização do treinamento dos modelos, foram aplicadas as seguintes técnicas de aumento de dados aleatoriamente através de uma camada da CNN: rotação e zoom de imagem em 20 graus e inversão vertical de imagem, A inversão vertical foi realizada pois um mesmo sinal em Libras pode ser efetuado com ambas as mãos. Através do uso destas técnicas foi possível notar um aumento de acurácia de cerca de 10% em todos os modelos treinados.

4. RESULTADOS

Ao total foram gerados 7 modelos, cada um usando uma das técnicas de pré-processamento de imagem proposta. Cada um dos modelos foram comparados entre si utilizando as métricas de acurácia, f1-score e tempo de treinamento. A curva de Loss gerada do treinamento também foi analisada. Os modelos gerados estão indexados na Tabela 2, estando também presente na mesma as métricas coletadas dos treinamentos.

4.1 Análise de métricas obtidas

Conforme exibido na Tabela 2, o modelo que melhor desempenhou em quesito de acurácia, tanto em no conjunto de treinamento, quanto validação, foi o modelo treinado com os dados originais normalizados com seus 90.08% de acurácia média, entretanto, vale destacar que este foi o modelo que mais demorou no treinamento. Outros que se destacaram no quesito acurácia, mas estando um pouco abaixo dos dados originais, foram os modelos treinados usando Canny, Sobel e Sobel+Otsu que apresentaram resultados próximos aos 80% em tempos de treinamento similares, sendo que o Sobel+OTSU aquele que obteve os melhores resultados deste grupo. Todos os outros apresentaram resultados abaixo dos 70% em acurácia no treinamento e teste, com exceção daquele treinado em níveis de cinza que mesmo com baixa acurácia no conjunto de treinamento desempenhou próximo aos 80% no conjunto de validação.

Tabela 2: Resultados de métricas de treinamento.

Modelo	Acurácia treinamento	Acurácia validação	F1-Score	Tempo de treinamento
Frames originais com normalização	86.74%	95.42%	0.509	40 minutos
Tons de cinza	67.46%	79.58%	0.505	30 minutos
Bordas Canny	78.08%	74.86%	0.538	25 minutos
Binarização adaptativa	49.65%	63.75%	0.388	32 minutos
Mapas de saliência	70.17%	80.69%	0.530	27 minutos
Bordas Sobel	78.09%	83.47%	0.505	27 minutos
Sobel+OTSU	81.02%	82.31%	0.541	28 minutos

Fonte: Autoria própria.

Com relação a métrica f1-score, percebeu-se que nenhum dos modelos treinados se sobressaiu em relação aos outros, onde todos ficaram próximos do valor 0.5, que é um valor não ideal, mas decente. Tendo como única exceção o modelo de binarização adaptativa, que está 0.2 pontos atrás dos demais.

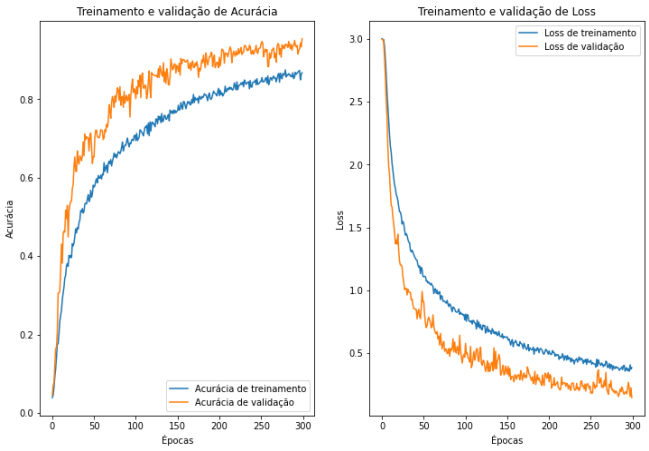
Quanto ao tempo de treinamento, os resultados foram unânimes, aqueles modelos que usaram algum pré-processamento de imagem treinaram consideravelmente mais rápido do que o que

usou os dados originais com normalização, sendo em média 11 minutos mais rápidos no treinamento das mesmas 300 épocas.

4.2 Análise da curva de Loss e validação

Algo que foi constante em todas as curvas de Loss geradas no treinamento dos modelos foi que o valor da curva de validação foram menores que os valores da curva de treinamento, isso se deve a utilização das camadas de dropout na CNN. Outro comportamento global foi a grande presença de ruído em todas as curvas obtidas, o que indica que o dataset utilizado possui um conjunto de validação não representativo, conforme pode ser visto na Figura 2, Figura 3 e Figura 4.

Figura 2: Curvas de treino e validação de modelo treinado com dados originais normalizados.



Fonte: Autoria própria.

Quando comparado às curvas de validação e treinamento de Loss de todos os modelos treinados, percebeu-se três comportamentos diferentes: o modelo treinado com os valores originais normalizados obteve a melhor curva, apresentando o menor ruído no decorrer das épocas e com a melhor capacidade de generalização dado a proximidade entre as curvas, conforme mostrado na Figura 2; já os modelos Canny, Saliency, Sobel, Gray e Sobel+OTSU, apresentaram curvas bem semelhantes, tendo uma boa proximidade entre as curvas, embora maior que o obtido com os dados originais, e também não apresentou indícios claros de overfitting e underfitting. conforme mostrado na Figura 3. E o último comportamento observado foi o da binarização adaptativa que apresentou sinais claros de Underfitting pela distância e formato entre as curvas de treino e validação, a presença de ruídos foi maior nesse gráfico, conforme mostrado na Figura 4.

4.3 Comparação com trabalhos semelhantes

Ao comparar a arquitetura dos modelos treinados com a de trabalhos semelhantes, nota-se que três se destacam em níveis de similaridade, portanto valem a pena serem comparados em nível de acurácia obtida a partir do conjunto de dados observados, sendo esses: o trabalho desenvolvido por Bheda el al. [13] que compartilha do uso de CNN como base arquitetural para frames estáticos do alfabeto da linguagem de sinais americana; o trabalho desenvolvido por Lazo *et al.* [14] que assim como o anterior usa CNN com uma abordagem semelhante para o alfabeto da linguagem de sinais peruana; e por fim o trabalho desenvolvido por Rezende *et al.* [17] que embora utilize vídeos com uma CNN-3D e não imagens, é um trabalho de estado da arte que usa o mesmo dataset que os modelos treinados neste trabalho. A Tabela 3 contém a comparação entre as acurácias médias obtidas pelos melhores modelos de cada trabalho.

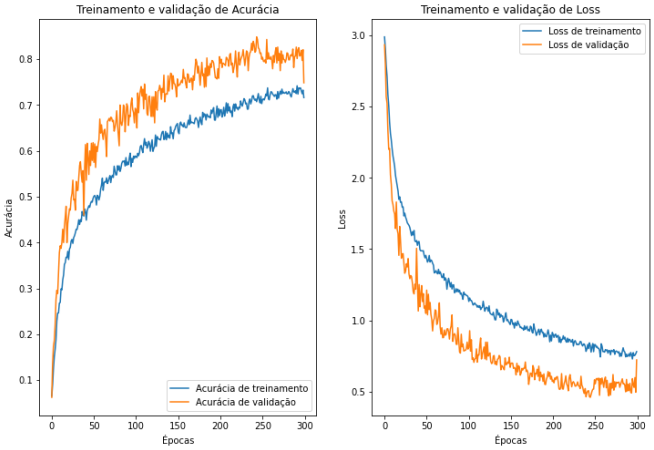
Tabela 3: Comparação de resultados com trabalhos semelhantes.

Modelo	Acurácia média	Tipo dataset
Modelo de Frames originais normalizados	91.08%	Frames extraídos do MNDIS-Libras
Bheda el al. [13]	89.75%	Dataset próprio com pouca diversidade
Lazo et al. [14]	99.85%	Dataset próprio com pouca diversidade
Rezende *et al.* [17]	93.03%	MNDIS-Libras

Fonte: Autoria própria.

Como é possível perceber ao observar a Tabela 3, o melhor modelo treinado neste trabalho, que foi utilizando dados originais normalizados, obteve uma acurácia média alta semelhante àquelas obtidas por trabalhos similares e equiparável com o trabalho estado da arte de Rezende el al. Agora é necessário destacar que o valores listados servem apenas para fins de comparação e não superioridade, pois as abordagens utilizadas nos trabalhos são diferentes com conjunto de dados simples, com exceção de Rezende *et al.* [17], o qual utiliza o mesmo conjunto de dados deste trabalho.

Figura 3: Curvas de treino e validação de modelo treinado com dados Sobel+OTSU.



Fonte: Autoria própria.

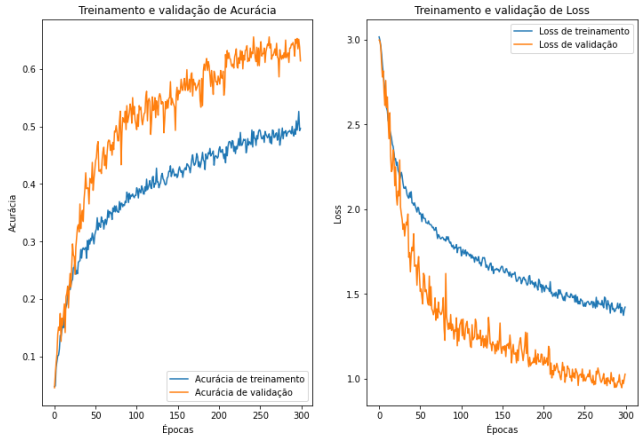
5. LIMITAÇÕES E TRABALHOS FUTUROS

Neste trabalho foi realizado um experimento para analisar os impactos do uso de diferentes técnicas de pré-processamento de imagens no treinamento de uma arquitetura CNN simples proposta, que resultou na criação de um modelo eficiente na classificação de um conjunto de 20 classes pré-definidas pertencentes a Libras utilizando os dados originais do conjunto de dados MINDS-Libras normalizados, com acurácia média de 90.08% e f1-score aceitável em torno dos 0.5. Foi possível notar também que o uso de técnicas de pré-processamento podem resultar em modelos com boa acurácia que conseguem ser treinados mais rápido do que aqueles que usam dados originais RGB normalizados.

Entretanto os resultados deste trabalho estão longe de poderem ser convertidos em soluções utilizáveis pela população brasileira de portadores de deficiência auditiva, que segundo o último censo brasileiro realizado pelo Instituto Brasileiro de Geografia e Estatística (IBGE) em 2010, totalizava cerca de 5% da população brasileira [7]. Dado o nicho da solução desenvolvida que foi testada em um único conjunto de dados, que embora seja grande, carece de dados semelhantes a cenários reais com grande variação de pose e iluminação. Portanto a testagem do modelo obtido com um conjunto de dados mais complexo e diverso seria um bom ponto de partida para continuação deste trabalho, possíveis limitações deste plano de expansão é a disponibilidade de conjunto de dados de Libras com este perfil.

Outro ponto que seria interessante desenvolver este trabalho, seria aplicar a arquitetura CNN utilizada para classificar imagens na classificação de vídeos utilizando CNN-3D em conjunto com um correlacionador temporal LTSM, semelhante à proposta por Stefano *et al.* [11]. Um possível impedimento para este plano de ação, é a disponibilidade de ambiente capaz de suportar o treinamento de vídeos com seus tempos de treinamentos altos.

Figura 4: Curvas de treino e validação de modelo treinado com dados de binarização adaptativa.



Fonte: Autoria própria.

6. AGRADECIMENTOS

Agradeço ao meu orientador por me auxiliar no processo de elaboração deste trabalho, principalmente através da sugestão de ideias e pela disponibilização de materiais de referência. Espero que possamos trabalhar juntos novamente em outro momento.

7. REFERÊNCIAS

[1] BRASIL. Lei nº 10.436, de 24 de abril de 2002. Dispõe sobre a Língua Brasileira de Sinais - Libras e dá outras providências. Brasília: Presidência da República, [2002]. Disponível em: http://www.planalto.gov.br/ccivil_03/leis/2002/l10436.htm. Acesso em 22 de março de 2022.

[2] BRASIL. Senado Federal. Projeto de Lei n° 6284, de 2019. Altera a Lei nº 9.394, de 20 de dezembro de 1996, que estabelece as diretrizes e bases da educação nacional, para estabelecer condições de oferta de ensino da Língua Brasileira de Sinais - LIBRAS, em todas as etapas e modalidades da educação básica. Brasília, DF: Senado Federal, 2019. Disponível em: https://www25.senado.leg.br/web/atividade/materias/-/materia/140061. Acesso em 22 de março de 2022.

[3] World Federation of the Deaf (WFD), "Position paper regarding the united nations convention on the rights of people with disabilities,". [Online] Disponível em: http://www.un.org/esa/socdev/enable/rights/contrib-wfd.htm. Acesso em 22 de março de 2022.

[4] GABRIELA, Ana. "LIBRAS é um idioma!"; IMEP. Disponível em: https://imepeducacional.com.br/libras-e-um-idioma/#:~:text=Importante%20saber%20que%20a%20L%C3%ADngua,24%20de%20abril%20de%202002. Acesso em 23 de março de 2022.

[5] BRASIL. Lei Nº 13.146, de 6 de julho de 2015. Institui a Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência). Brasília: Presidência da República, [2015]. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13146.htm. Acesso em 23 de março de 2022.

[6] BRASIL. IBGE. Pesquisa Nacional de Saúde - PNS 2019, 2019. Disponível em: <www.ibge.gov.br>. Acesso em 23 de março de 2022.

[7] BRASIL. IBGE. "Censo demográfico 2010 - características gerais da população, religião e pessoas com deficiência,", 2010. Disponível em: https://biblioteca.ibge.gov.br/visualizacao/periodicos/94/cd_2010_religiao_deficiencia.pdf. Acesso em 23 de março de 2022.

[8] Yauri Vidalón, J.E., De Martino, J.M. (2016). Brazilian Sign Language Recognition Using Kinect. In: Hua, G., Jégou, H. (eds) Computer Vision – ECCV 2016 Workshops. ECCV 2016. Lecture Notes in Computer Science(), vol 9914. Springer, Cham. https://doi.org/10.1007/978-3-319-48881-3_27

[9] W. C. Stokoe. Sign language structure: an outline of the visual communication systems of the american deaf. 1960. Journal of deaf studies and deaf education, 10 1:3–37, 2005.

[10] K. Nakamura. About american sign language. Deaf Resource Library, 1995.

[11] Stefano, Gabriel & Lobato Passos, Wesley & Gois, Jonathan & Araujo, Gabriel & Lima, Amaro. (2021). Um sistema de reconhecimento de sinais em Libras usando CNN e LSTM. 10.14209/sbrt.2021.1570727292.

[12] Kayo Yin. Sign language translation with transformers. arXiv preprint arXiv:2004.00588, 2020.

[13] Bheda, Vivek & Radpour, Dianna. (2017). Using Deep Convolutional Networks for Gesture Recognition in American Sign Language.

[14] Lazo Quispe, Cristian & Sanchez, Zaid & Carpio, Christian. (2019). A Static Hand Gesture Recognition for Peruvian Sign Language Using Digital Image Processing and Deep Learning. 10.1007/978-3-030-16053-1_27.

[15] Sharma, Ashish & Mittal, Anmol & Singh, Savitoj & Awatramani, Vasudev. (2020). Hand Gesture Recognition using Image Processing and Feature Extraction Techniques. Procedia Computer Science. 173. 181-190. 10.1016/j.procs.2020.06.022.

[16] Barros, Pablo & Magg, Sven & Weber, Cornelius & Wermter, Stefan. (2014). A Multichannel Convolutional Neural Network for Hand Posture Recognition. 403-410. 10.1007/978-3-319-11179-7_51.

[17] T. Rezende, S. Almeida, and F. Guimarães, “Development and validation of a brazilian sign language database for human gesture recognition,” Neural Computing and Applications, vol. 1, pp. 1–19, Mar. 2021.

[18] N. Otsu, “A threshold selection method from gray-level histograms,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[19] Microsoft Inc. Kinect for Windows SDK 2.0. (2014). https://developer.microsoft.com/en-us/windows/kinect/develop. Accessed 19 Aug 2016