Detecção de Linguagem de Sinais usando Redes Neurais Convolucionais

Henrique Andrade Lopes Jonas Magalhães Moreira Samuel Raimundo Lopes Pinto

I. Introdução

A comunicação é um dos aspectos mais fundamentais da humanidade, essencial para a interação social e a transmissão de conhecimento. Nesse contexto, a Língua Brasileira de Sinais (Libras) emergiu como uma necessidade vital para a inclusão das pessoas surdas no Brasil [1]. Libras proporciona um meio eficaz de comunicação e educação para a comunidade surda, permitindo sua plena participação na sociedade, acesso à educação e ao mercado de trabalho, além de fortalecer sua identidade cultural e nacional.

Contudo, a Libras ainda não é amplamente conhecida pela maioria da população brasileira sem deficiência auditiva. Essa lacuna na comunicação cria barreiras significativas para a inclusão social das pessoas surdas [2]. Portanto, a facilitação da comunicação entre indivíduos que utilizam Libras e aqueles que não a dominam torna-se uma demanda crucial para promover a inclusão plena na sociedade.

Diante desse desafio, este trabalho propõe uma abordagem baseada em Redes Neurais Convolucionais (CNN) para detectar e interpretar automaticamente a língua de sinais. Através de uma abordagem de aprendizado supervisionado, buscamos desenvolver um sistema eficiente que promova a acessibilidade e a comunicação entre surdos e ouvintes.

II. Trabalhos Relacionados

Muitos trabalhos abordam o reconhecimento de gestos de mão com deep learning. A categorização dos gestos é dividida em reconhecimento estático e dinâmico [3]. Nosso trabalho foca especificamente no reconhecimento estático, que envolve a análise de gestos realizados e mantidos sem mudanças significativas ao longo do tempo.

Quando olhamos para trabalhos similares de reconhecimento estático de linguagem gestual, observamos uma variedade de abordagens. Por exemplo, Kong Y et al. [4] extraíram cor, profundidade e contorno da imagem para reconhecer a linguagem gestual. Zhang et al [5] propuseram um método de detecção de linguagem gestual americana baseado em Resnet-18 e aumento de dados, alcançando uma precisão média de 99% na identificação de soletração manual. Além disso, também é comum a utilização do modelo VGG16, como feito por Tanseem N. Abu-Jamie et al [6] para melhorar a extração de características visuais na classificaçao de imagens.

Nesse trabalho buscamos unir essas diferentes estratégias já muito utilizadas no campo do reconhecimento de gestos de mão.

III. Metodologia

A. Dataset

Neste trabalho, utilizamos um conjunto de dados composto por imagens do alfabeto da Língua de Sinais Americana (ASL)[1]. O dataset original é organizado em 29 pastas, cada uma representando uma classe diferente. No entanto, excluímos as classes ”Nothing”, ”Space” e ”Delete” por não fazerem parte do alfabeto que é nosso foco. Além disso, optamos por remover as letras ”J” e ”Z”, uma vez que estas requerem movimento e nosso estudo se concentra apenas em símbolos estáticos. Para reduzir o volume total de imagens e possibilitar a execução dos testes em tempo hábil, selecionamos aleatoriamente as letras ”P”, “Q” e “X” para exclusão.

As imagens restantes consistem em aproximadamente 8000 exemplares para cada letra do alfabeto, capturadas de diferentes indivíduos e em diversos ambientes e condições. Cada imagem tem uma resolução de 200x200 pixels. A Figura 1 ilustra o alfabeto da Língua de Sinais Americana utilizado neste estudo.

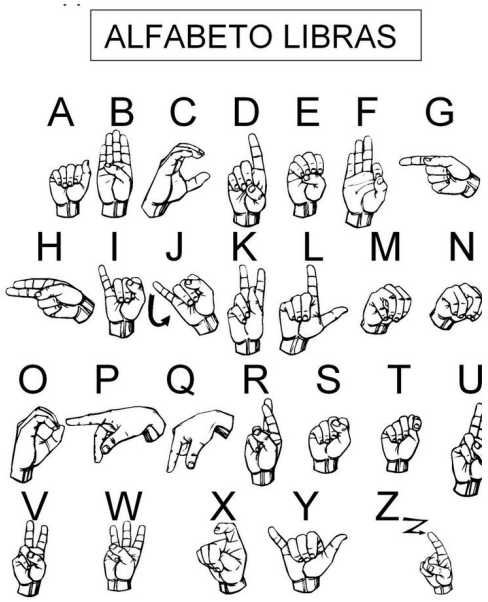

Fig. 1. Alfabeto Libras. Fonte: https://br.pinterest.com/pin/592575263470244217/

[1]https://www.kaggle.com/datasets/debashishsau/aslamerican-sign-language-aplhabet-dataset

B. Redes propostas

Para a classificação de sinais de mão do alfabeto americano, foram utilizadas duas arquiteturas diferentes de Redes Neurais Convolucionais (CNN). A Figura 2 apresenta a primeira arquitetura proposta, denominada INF692NET, adaptada a partir da rede descrita em [7], a qual foi treinada do zero. Já a Figura 3 exibe a arquitetura da ResNet18, que passou por um processo de fine-tuning em todas as camadas, utilizando pesos pré-treinados no conjunto de dados ImageNet.

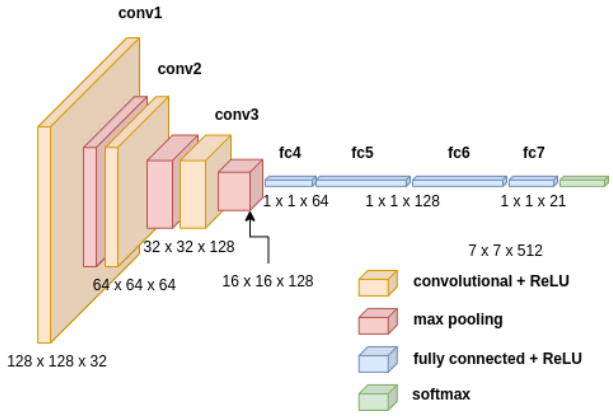

Fig. 2. Arquitetura INF692NET

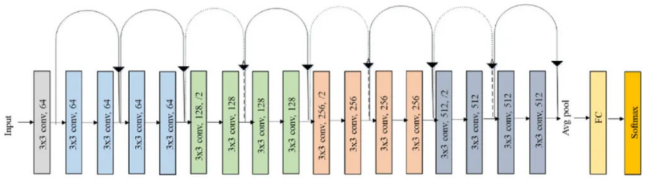

Fig. 3. Arquitetura ResNet18. Fonte: https://www.kaggle.com/datasets/debashishsau/aslamerican-sign-language-aplhabet-dataset

C. Design Experimental

Os dados coletados foram divididos em conjuntos de treino e teste, com 161.276 imagens destinadas para o treinamento e 6.300 imagens selecionadas aleatoriamente para teste, correspondendo a 300 imagens por classe.

Para a utilização no modelo INF692NET, inicialmente redimensionamos as imagens para 128x128 pixels, visando facilitar o processamento em lotes. Em seguida, as imagens foram transformadas de uma matriz de pixels para um tensor, formato adequado para o processamento em uma rede neural. Finalmente, normalizamos as imagens utilizando médias de 0.4914, 0.4822 e 0.4465, e desvios padrão de 0.2023, 0.1994 e 0.2010 para as cores vermelho, verde e azul, respectivamente. A normalização é crucial para estabilizar e acelerar o treinamento, ajudando a rede a convergir mais rapidamente ao evitar grandes variações nos valores dos pixels.

Para o modelo ResNet-18, utilizamos transformações padronizadas aplicadas durante o pré-treinamento no ImageNet. Carregamos os pesos pré-treinados do dataset ImageNet, aproveitando o fato de a ResNet-18 possuir uma arquitetura bem estabelecida, projetada para resolver problemas de degradação de rede através de blocos residuais. Estes blocos permitem que as camadas mais profundas contribuam efetivamente para a extração de características [8] . Além disso, o pré-treinamento fornece conhecimento prévio sobre diversas características visuais de baixo e alto nível, como bordas, texturas, formas e objetos [9]. Utilizamos uma rotina de Fine Tuning, retreinando todas as camadas da rede e substituindo o classificador final para que ele fosse responsável por determinar uma entre as 21 classes definidas no contexto do problema.

A seguir, apresentam-se as configurações experimentais utilizadas para cada um dos modelos.

1) INF692NET:

- Épocas: 55
- Otimizador: ADAM
- Taxa de aprendizado: 0.001
- Ambiente de Treinamento:
  - Processador: AMD Ryzen 7 5700X 8-Core (3.40 GHz)
  - Memória RAM: 16 GB
  - GPU: NVIDIA GeForce RTX 3060 com 12 GB de memória dedicada
- Tempo de Treinamento: $\sim 6horas$

2) ResNet-18:

- Épocas: 10
- Otimizador: ADAM
- Taxa de aprendizado: 0.001
- Ambiente de Treinamento:
  - Processador: Intel Core I5-10210U (1.60 GHz x 8)
  - Memória RAM: 16 GB
  - GPU: NVIDIA MX110
- Tempo de Treinamento: $\sim 15horas$

IV. Resultados

A. INF692NET

A Figura 4 mostra os gráficos de perda e acurácia do treinamento do modelo INF692NET. É possível observar que o treinamento foi bem-sucedido, já que a função de perda ao longo das épocas apresenta uma diminuição constante, evidenciando o aprendizado progressivo do modelo.

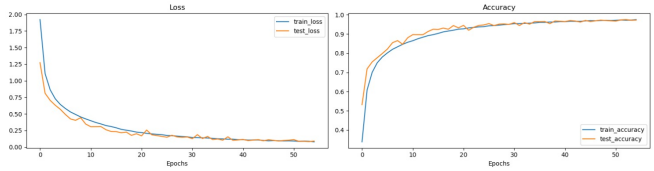

Fig. 4. Gráficos de perda e acurácia do modelo INF692NET

A matriz de confusão do modelo INF692NET, apresentada na Figura 5, nos permite visualizar o desempenho em termos de classificações corretas e incorretas. Esta matriz foi gerada utilizando 300 imagens de teste para cada classe. Analisando

os resultados, é possível notar que o modelo teve um bom desempenho, com um baixo número de erros. No entanto, um erro comum foi observado na letra ”W”, que o modelo tende a classificar como a letra ”V” em uma quantidade pequena, mas significativa, das vezes, devido à semelhança entre os dois símbolos.

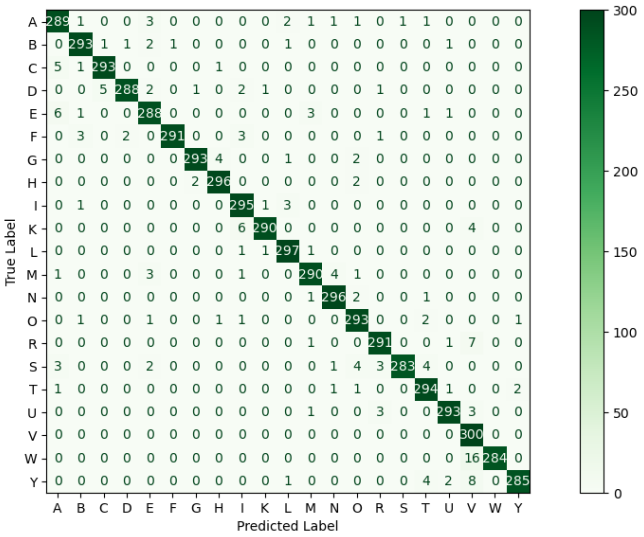

Fig. 5. Matriz de Confusão do modelo INF692NET

B. ResNet18

Os gráficos de perda e acurácia ao longo do Fine Tuning da ResNet18 são demonstrados na Figura 6. Embora os valores pareçam variar mais, isso se deve à escala utilizada, já que a acurácia começa alta e a perda baixa, devido ao fato dos pesos já serem inicializados com o treinamento da rede no conjunto de dados ImageNET. Esses gráficos mostram que o modelo ResNet18 manteve um bom desempenho ao longo do treinamento, com uma acurácia elevada e uma função de perda relativamente estável, tendendo a zero. Nota-se que a ResNet18 precisou de apenas 10 épocas para alcançar resultados consideráveis, com uma perda final menor que a do modelo INF692NET.

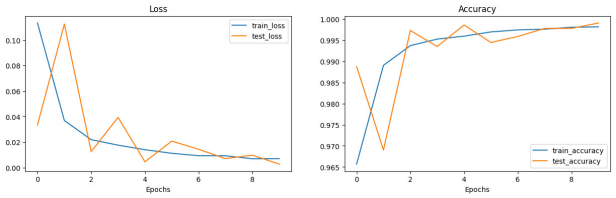

Fig. 6. Gráficos de perda e acurácia do modelo ResNet18

A matriz de confusão da ResNet18, apresentada na Figura 7, evidencia ainda mais o desempenho superior do modelo em comparação ao INF692NET, apresentando apenas 6 erros para o conjunto de teste utilizado.

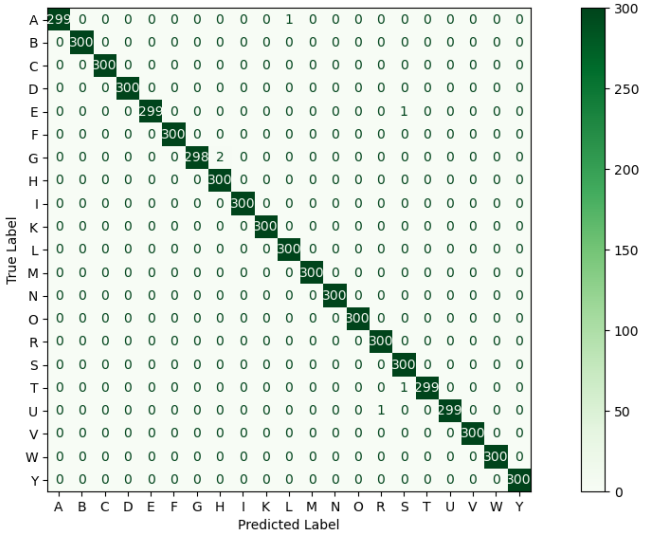

Fig. 7. Matriz de Confusão do modelo ResNet18

C. Preditor em tempo real

Por fim, após o treinamento dos modelos, integramos um sistema que captura fotos em tempo real e as envia para os modelos, que retornam qual letra do alfabeto americano o gesto representa. A interface do sistema é simples e intuitiva: ao abrir a tela, qualquer sinal de mão capturado pela câmera é interpretado pelos modelos, que exibem em tempo real a letra identificada.

A figura 8 ilustra o resultado dessa integração utilizando o modelo INF692NET.

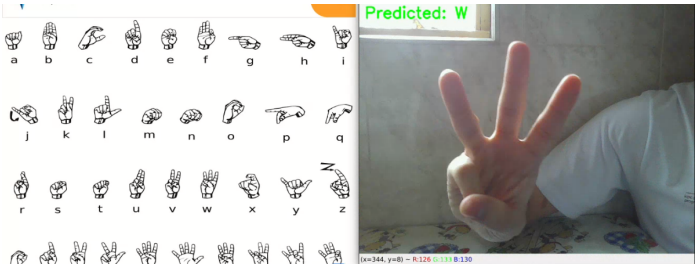

Fig. 8. Exemplo de predição

V. Conclusão

Em suma, este trabalho explorou a aplicação de Redes Neurais Convolucionais para o reconhecimento automático de sinais de mão do alfabeto da Língua de Sinais. Utilizando as arquiteturas INF692NET e ResNet-18, demonstramos que ambas são capazes de alcançar resultados significativos na classificação de gestos estáticos. Através de experimentos detalhados e análises dos modelos treinados, verificamos que a ResNet-18, com seu processo de Fine Tuning e utilização de pesos pré-treinados do ImageNet, obteve um desempenho superior, mostrando menor perda e maior acurácia em comparação ao INF692NET.

Além disso, a integração bem-sucedida de um preditor em tempo real reforça a viabilidade prática desses modelos, oferecendo uma ferramenta potencial para melhorar a acessibilidade e a comunicação para indivíduos surdos. Este estudo não apenas contribui para o avanço no reconhecimento de gestos de mão, mas também destaca a importância das redes neurais convolucionais na promoção da inclusão social através da tecnologia.

REFERENCES

[1] V. M. Fernandes, "A importância da comunicação em libras para o surdo brasileiro," 2018.

[2] M. F. N. S. d. Souza, A. M. B. Araújo, L. F. F. Sandes, D. A. Freitas, W. D. Soares, R. S. d. M. Vianna, and Á. A. D. d. Sousa, "Principais dificuldades e obstáculos enfrentados pela comunidade surda no acesso à saúde: uma revisão integrativa de literatura," *Revista Cefac*, vol. 19, pp. 395–405, 2017.

[3] A. A. Barbhuiya, R. K. Karsh, and R. Jain, "Cnn based feature extraction and classification for sign language," *Multimedia Tools and Applications*, vol. 80, no. 2, pp. 3051–3069, 2021.

[4] Y. Kong, B. Satarboroujeni, and Y. Fu, "Learning hierarchical 3d kernel descriptors for rgb-d action recognition," *Computer Vision and Image Understanding*, vol. 144, pp. 14–23, 2016, individual and Group Activities in Video Event Analysis. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314215002118

[5] Z. Han-wen, H. Ying, Z. Yong-jia, and W. Cheng-yu, "Fingerspelling identification for american sign language based on resnet-18," *International Journal of Advanced Networking and Applications*, vol. 13, no. 1, pp. 4816–4820, 2021.

[6] T. N. Abu-Jamie and S. S. Abu-Naser, "Classification of sign-language using vgg16," 2022.

[7] R. Patil, V. Patil, A. Bahuguna, and G. Datkhile, "Indian sign language recognition using convolutional neural network," *ITM Web of Conferences*, vol. 40, p. 03004, 08 2021.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[9] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint arXiv:1608.08614*, 2016.