Uso de Técnicas de Deep Learning para o Reconhecimento da Língua de Sinais

André Luiz Santos Moreira da Silva[1], **Daniel Rocha Franca**[1], **Ligia Ferreira de Carvalho Gonçalves**[1]

[1]Instituto de Ciências Exatas e Informática – Pontifícia Universidade Católica de Minas Gerais (PUC Minas) – 30140-100 – Belo Horizonte – MG – Brazil

Abstract.

Resumo.

1. Contextualização

A comunicação é um dos pilares essenciais da vida humana. Nossa espécie se destaca pela sociabilidade, uma característica intrínseca que precede, em muitos aspectos, o pleno desenvolvimento da racionalidade complexa. Dentro desse contexto social, a fala emerge como nosso principal método de interação. Contudo, a comunicação vai muito além da fala; ela se manifesta de diversas maneiras, e para uma parcela considerável da população, essa expressão se dá por meio da língua de sinais.

A língua de sinais é uma esfera universal em que cada país possui sua própria variação. A Organização Mundial de Surdos , dentre os 70 milhões de surdos, coletivamente existam mais de 300 línguas de sinais distintas. Como exemplo, podemos citar a ANL (American Sign Language), LIBRAS (Língua Brasileira de Sinais), dentre outras. De acordo com **?**, a interpretação dos sinais depende de cinco elementos: o formato das mãos, as expressões faciais, a orientação da palma, movimentações, a localização e as expressões faciais. Sistemas de reconhecimento de sinais não são uma linha de pesquisa recente. Em 1986, **?** propuseram uma interface que capturava informações em tempo real sobre gesticulações utilizando as mãos. Apesar dos recentes avanços, esses sistemas ainda apresentam limitações devido à forte influência de variáveis externas sobre seu desempenho, como iluminação, posicionamento da máquina e até mesmo como o indivíduo construiu determinado sinal, a principal sendo a capacidade de traduzir frases completas em tempo real.

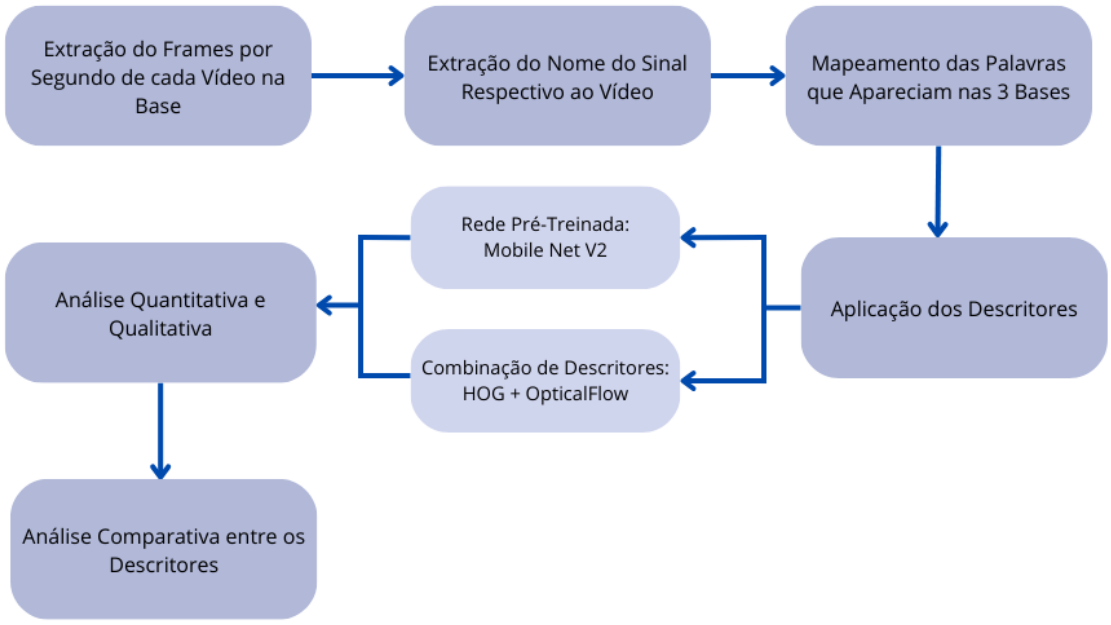
Desse modo, a viabilização de métodos e processos responsáveis pelo reconhecimento desses sinais é fundamental para a manutenção do diálogo entre indivíduos. Suas aplicações ultrapassam o contexto da comunicação, estendendo-se a diversas áreas. Por exemplo, na educação, sistemas de reconhecimento de sinais podem facilitar o aprendizado de pessoas surdas, promovendo a inclusão em ambientes educacionais. Em serviços públicos, essas tecnologias garantem que informações e serviços governamentais sejam acessíveis a todos os cidadãos, independentemente de suas habilidades auditivas. Na saúde, a comunicação entre profissionais de saúde e pacientes surdos é aprimorada, assegurando um atendimento mais eficaz e humanizado. Além disso, no mercado de trabalho, ampliam-se as oportunidades de emprego para pessoas surdas, ao tornar ambientes corporativos mais inclusivos e adaptados às suas necessidades. Conforme destacado no estudo **?**, a surdez tem um impacto negativo sobre a qualidade de vida relacionada à saúde

(QVRS) de pessoas surdas, sendo que sintomas de ansiedade e depressão são mais acentuados nesse grupo e podem estar relacionados a dificuldades de comunicação. Portanto, investir em tecnologias e processos que aprimorem o reconhecimento e a interpretação das línguas de sinais é essencial para promover a inclusão social e garantir que indivíduos surdos possam participar plenamente de todas as esferas da sociedade.

2. Estudo e Comparação de Embeddings

O dataset separado para a realização deste trabalho, que está disponível em (https://libras.cin.ufpe.br/), é estruturado da seguinte maneira: três articuladores diferentes possuem um vídeo associado a cada um, representando o sinal de uma palavra. Ao todo, a base apresenta 1364 palavras, totalizando, com três bases, 4089 ocorrências. No entanto, durante os primeiros trabalhos com o dataset, algumas inconsistências foram descobertas; por exemplo, algumas palavras não estavam presentes em todas as bases. Essas questões precisaram ser tratadas.

Figure 1. Esquema Descritivo das Etapas do Processo de Ensemble



Desse modo, tendo como base a Figura 1, é possível abordar cada etapa de forma detalhada:

Extração dos Frames por Segundo de Cada Vídeo na Base: Para essa etapa, foi utilizada a biblioteca OpenCV, cuja principal característica é a leitura e exibição de imagens e vídeos, possibilitando, por isso, a captura e manipulação tanto em tempo real quanto de arquivos gravados. Dessa forma, foi usada para capturar um frame por segundo de cada vídeo. Após isso, foi criada uma subpasta para armazenar os frames de cada vídeo. O código pode ser encontrado em: Notebook de Extração de Frames

Extração do Nome do Sinal Respectivo ao Vídeo: Após a extração dos vídeos, foi usada a biblioteca Tesseract, que é uma ferramenta de reconhecimento óptico de caracteres (OCR), ou seja, uma API capaz de reconhecer caracteres a partir de arquivos de

imagem. Ela foi utilizada para remover os textos contidos em cada vídeo, permitindo, assim, que cada 'classe' fosse nomeada. O código pode ser encontrado em: Notebook Extração Nome Sinal

2.1. Implementação de Abordagens

A abordagem de combinar descritores para a extração de características não é novidade. No trabalho de **?**, os autores utilizaram a combinação dos descritores HOG (Histogram of Oriented Gradients) e HOF (Histogram of Optical Flow) para desenvolver um método de reconhecimento de gestos baseado em aprendizado a partir de um único exemplo (one-shot learning). Já em **?**, os autores propuseram a concatenação de tensores individuais calculados com Optical Flow Approximation e HOG3D, formando um descritor global final utilizado para o reconhecimento de ações humanas. Essa abordagem se mostrou eficiente, uma vez que apresentou boas taxas de reconhecimento baseadas exclusivamente em vídeo.

O uso de redes pré-treinadas para extração de características de imagem é uma prática consolidada, amplamente adotada na literatura científica. Em trabalhos mais recentes, como o de **?**, os autores propõem a fusão entre uma rede CNN tradicional e módulos baseados em Transformers, com o objetivo de extrair simultaneamente características locais e globais das imagens. Os resultados experimentais demonstram que a estrutura de rede híbrida contribuiu significativamente para a melhoria da acurácia no reconhecimento de gestos. Dessa forma, é possível afirmar que os dois métodos adotados neste trabalho possuem fundamentação teórica consistente, o que justifica suas respectivas escolhas.

Combinação de Descritores:

Neste experimento, o objetivo é combinar descritores extraídos de três bases de vídeos em LIBRAS, organizadas em diretórios. Para isso, utilizamos a rede neural pré-treinada ResNet50, que é especializada na extração de características de imagens. O modelo ResNet50, ao ser aplicado, gera um vetor de características para cada imagem, capturando informações relevantes, como formas, texturas e padrões. A extração dos descritores é feita a partir de até três frames por sequência de vídeo, escolhendo as imagens de cada diretório e aplicando a rede para gerar os vetores representativos de cada uma.

A abordagem adotada para a extração dos descritores inclui o uso de uma função personalizada que percorre os diretórios das três bases de dados, carrega as imagens, aplica o pré-processamento necessário (como redimensionamento e normalização) e então extrai os descritores usando o modelo ResNet50. Cada imagem é processada individualmente, e seu descritor resultante é armazenado em uma lista. Essa abordagem permite que todas as imagens sejam tratadas de forma independente, garantindo que o descritor de cada uma seja calculado e armazenado corretamente para futura análise.

Após a extração dos descritores, todos os vetores são combinados em um único array, onde cada linha representa os descritores de uma imagem. Esse array combinado é então salvo em um arquivo CSV, o que facilita o uso posterior dos dados em outros processos de análise, como visualização com técnicas de redução de dimensionalidade ou treinamento de modelos de aprendizado de máquina. O processo de combinação de descritores permite a criação de uma representação global das amostras das três bases,

facilitando a análise comparativa entre elas e demonstrando a efetividade da ResNet50 na captura de características relevantes das imagens de LIBRAS.

Descritor Baseado em uma Rede Neural: Para realizar a tarefa da descrição dos frames extraídos dos vídeos, escolhemos a MobileNetV2, uma rede convolucional utilizada em aplicações para dispositivos móveis, sendo, portanto, leve e prática. A MobileNetV2 é construída para lidar com tarefas envolvendo o processamento de imagens, algo demonstrado em seu trabalho de origem **?**, onde é testada e treinada com tarefas de visão computacional. Neste trabalho, buscamos a versão da MobileNetV2 que a biblioteca TensorFlow fornece, que segue a estrutura original, porém nos permite usar os pesos da rede treinada na base ImageNet, amplamente usada na visão computacional. Como desejamos apenas a habilidade de descrever os frames com a MobileNetV2 treinada na ImageNet, removemos as camadas de classificação da rede treinada, obtendo como resultado uma saída 3D que descreve os mapas de ativações que a rede obteve para aquele frame específico. Em seguida, tratamos essa saída adicionando à rede uma camada de Global Average Pooling 2D, para transformá-la em um vetor. Após aplicarmos o descritor sobre cada frame de cada vídeo, combinamos os vetores resultantes em uma matriz que representa todos os frames do vídeo. Após concluirmos a extração da descrição de cada frame, realizamos uma avaliação da qualidade dos embeddings gerados. Para isso, utilizamos três abordagens complementares: cálculo da similaridade intra e interclasse, visualização com t-SNE e classificação com regressão logística. Nos próximos tópicos vamos explorar cada abordagem realizada.

2.1.1. Análise da Qualidade dos Embeddings Rede Neural

Na primeira abordagem, calculamos a similaridade média dos vetores entre vídeos da mesma classe (intra-classe) e entre vídeos de classes diferentes (inter-classe), utilizando a métrica de similaridade cosseno. O resultado obtido foi uma similaridade média intra-classe de 0.7936 e uma similaridade média inter-classe de 0.8079, indicando uma proximidade elevada entre vídeos, independentemente da classe, o que pode sugerir que os embeddings não estão separando bem as categorias semânticas.

2.1.2. Visualização Rede Neural

Já na intenção de visualizar, aplicamos o algoritmo t-SNE para reduzir os vetores a duas dimensões e visualizar os agrupamentos no espaço vetorial. A visualização gerada (Figura **??**) mostra alguns agrupamentos visuais aparentes, mas também uma considerável sobreposição entre diferentes classes, o que reforça a hipótese de que o descritor ainda não está totalmente conseguindo capturar bem as diferenças entre as categorias.

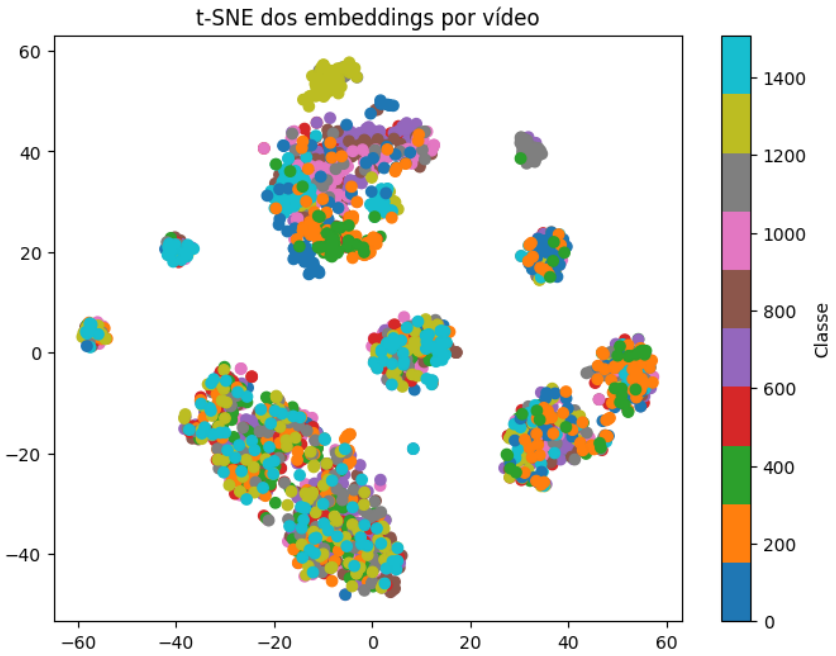


Figure 2. Visualização dos embeddings por vídeo com t-SNE. Cada ponto representa um vídeo, colorido conforme sua classe.

2.1.3. Aplicação utilizando os Embeddings Rede Neural

Treinamos um modelo de regressão logística simples com os vetores extraídos como entrada. A acurácia obtida no conjunto de teste foi de apenas 0.0039, o que pode confirmar que os vetores gerados ainda não carregam informações discriminativas suficientes. Tornando difícil a classificação supervisionada eficaz sem deep learning.

2.1.4. Aplicação utilizando os Embeddings Combinação de Descritores

A proposta de combinação de descritores tem como objetivo capturar informações complementares sobre os vídeos, unindo características espaciais e temporais. Para isso, foram utilizados dois tipos de descritores tradicionais: o Histogram of Oriented Gradients (HOG) e o Optical Flow. O HOG é responsável por representar a estrutura espacial das imagens, por meio da análise de contornos e bordas, enquanto o Optical Flow, obtido pelo método de Farneback, registra o movimento de pixels entre frames consecutivos, fornecendo informações sobre a dinâmica da ação no vídeo.

Cada um desses descritores foi extraído individualmente a partir dos frames dos vídeos, sendo em seguida normalizados para garantir comparabilidade entre os vetores. Após a normalização, os vetores foram concatenados horizontalmente, formando um único vetor representativo por vídeo, que integra tanto os aspectos espaciais quanto os temporais. Considerando o alto número de variáveis geradas, foi aplicada uma etapa de redução de dimensionalidade utilizando Análise de Componentes Principais (PCA), com o objetivo de comprimir os dados, eliminar redundâncias e tornar os vetores mais compactos para análise subsequente. Essa abordagem permitiu obter embeddings combi-

nados que representam, de maneira unificada, as principais características observáveis no conteúdo visual dos vídeos.

2.1.5. Visualização Combinação de Descritores

Com os embeddings combinados gerados a partir dos descritores HOG e Optical Flow, foi realizada uma visualização dos dados utilizando a técnica de t-Distributed Stochastic Neighbor Embedding (t-SNE). Essa técnica é amplamente utilizada para redução de dimensionalidade e projeção de dados de alta dimensão em um espaço bidimensional, de modo a facilitar a interpretação visual dos agrupamentos formados.

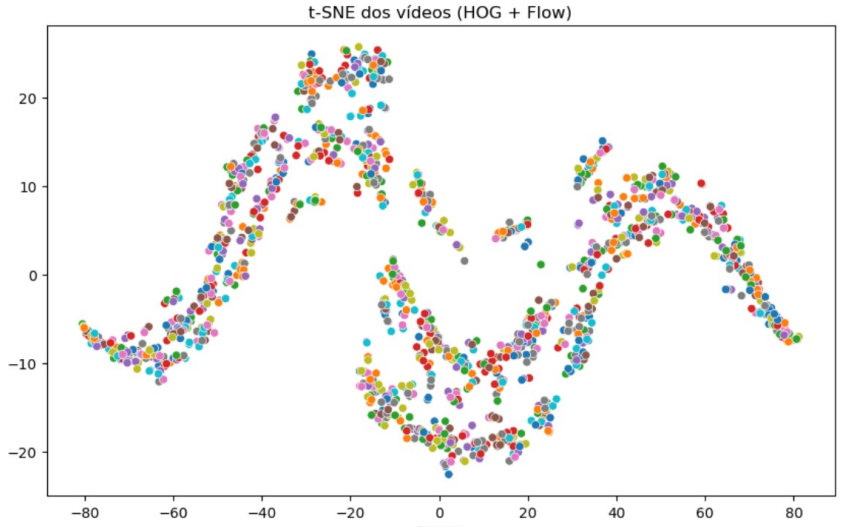


Figure 3. Visualização dos embeddings por HOG x Opticaw Flow com t-SNE.

A projeção dos dados revelou a existência de alguns agrupamentos visuais que indicam padrões de similaridade entre vídeos. No entanto, observou-se uma significativa sobreposição entre as classes, o que dificulta a distinção clara entre os diferentes grupos. Tal sobreposição sugere que, mesmo após a combinação de descritores e a aplicação de PCA, os vetores gerados não conseguem capturar suficientemente as diferenças intrínsecas entre as classes representadas nos vídeos. Esse resultado é consistente com a hipótese de que a fusão de descritores tradicionais, embora rica em informações, não é suficiente por si só para garantir uma separação clara entre categorias complexas de ações visuais.

2.1.6. Análise da Qualidade dos Embeddings Combinação de Descritores

A qualidade dos embeddings obtidos pela combinação de descritores foi avaliada por meio de métricas quantitativas e análises qualitativas. Inicialmente, foi utilizada a métrica de similaridade cosseno para mensurar a proximidade entre os vetores de uma mesma classe (intra-classe) e de classes diferentes (inter-classe). Os resultados demonstraram uma média de similaridade intra-classe de -0.0177 e inter-classe de -0.0005, indicando

baixa separabilidade entre os grupos. Valores tão próximos de zero — e até mesmo negativos — sugerem que os vetores não estão organizados de forma consistente em relação às classes, o que compromete sua utilidade como representações discriminativas.

Complementarmente, foi realizada uma avaliação supervisionada utilizando regressão logística para classificar os vídeos a partir dos embeddings. A acurácia obtida no teste foi de apenas 0.2105, um desempenho muito abaixo do esperado para um classificador simples, o que reforça a dificuldade de separar corretamente as amostras com base nesses vetores. Esse baixo desempenho evidencia que, embora a fusão de HOG e Optical Flow combine diferentes perspectivas da informação visual, os embeddings resultantes não são suficientemente informativos para realizar uma classificação precisa entre classes distintas. Isso indica a necessidade de abordagens mais robustas ou do uso de descritores mais avançados, possivelmente baseados em modelos de aprendizado profundo.

3. Treinamento do Modelo

3.1. Treinamento do Modelo

Esta etapa tem como objetivo o treinamento de uma rede LSTM para a tarefa de reconhecimento de sinais da Língua Brasileira de Sinais (LIBRAS), utilizando como entrada os vetores de keypoints extraídos com o MediaPipe Holistic. A configuração atual do conjunto de dados foi definida a partir das análises conduzidas na etapa anterior, dedicada à extração e exploração de embeddings. Naquela fase, observou-se uma baixa separabilidade entre classes e uma elevada variabilidade intra-classe, o que motivou uma subseleção criteriosa das categorias mais consistentes e a coleta de novas instâncias com melhor qualidade visual.

Durante o treinamento, diferentes configurações de hiperparâmetros foram avaliadas de forma sequencial, com base no desempenho obtido em validação. O processo foi conduzido de maneira iterativa, com ajustes sucessivos visando à melhoria da acurácia do modelo, permitindo refinar gradualmente o comportamento da rede ao longo dos ciclos de teste.

3.2. Seleção de Classes

A definição das classes utilizadas no presente experimento foi fundamentada no estudo "A Cross-Dataset Study on the Brazilian Sign Language Translation" (Sarmento Ponti, 2023), apresentado no workshop da ICCVW. Esse trabalho teve como principal contribuição a padronização de um benchmark para tarefas de classificação de sinais em LIBRAS, com ênfase na avaliação da capacidade de generalização dos modelos em cenários realistas.

No referido estudo, os autores integraram quatro bases de dados distintas: V-LIBRASIL (UFPE), LIBRAS-Português (UFV), Dicionário de LIBRAS (INES) e SignBank (UFSC). Cada base possui características próprias quanto ao número de vídeos, variedade de articuladores e qualidade de gravação, o que impôs o desafio de selecionar um subconjunto de sinais que fosse comum a todas as fontes.

Inicialmente, foram identificadas 49 categorias que estavam presentes em todos os conjuntos e que possuíam, no mínimo, quatro instâncias por base. No entanto, observou-se que, mesmo entre sinais com o mesmo rótulo, havia significativa variação na execução

— reflexo de diferenças regionais, estilo individual e qualidade dos vídeos. Para mitigar esse ruído semântico e visual, foi realizada uma análise de variabilidade intra-classe, a partir da qual as categorias com maior inconsistência foram descartadas.

Após esse refinamento, chegou-se a um conjunto mais robusto de 37 classes, selecionadas com base na estabilidade visual dos sinais e na representatividade das instâncias. Essa curadoria garantiu maior homogeneidade nas amostras e favoreceu a avaliação mais justa do modelo.

No presente trabalho, foram utilizadas exclusivamente instâncias do Articulador 2 da base V-LIBRASIL (UFPE), totalizando três vídeos por classe. Complementarmente, foi incluída uma instância por classe proveniente das bases auxiliares CEAD (UFV) e Signbank - UFSC, totalizando 185 amostras. Essa escolha teve como objetivo explorar variações interbase de forma controlada, mantendo a coerência com o benchmark proposto no estudo de referência.

Table 1. Tabela de instâncias utilizadas no treinamento

Base	Origem	Instâncias por classe	Total de instâncias
V-LIBRASIL (Articulador 2)	UFPE	3	111
CEAD	UFV	1	44
Signbank	UFSC	1	37
Total	—	1 a 3	37

Entre as classes selecionadas, encontram-se sinais que representam objetos concretos, ações do cotidiano e conceitos diversos, tais como: acordar, amigo, bicicleta, coelho, comer, comparar, escola, esquecer, flor, melancia e patins, entre outros. A escolha dessas categorias visa avaliar a capacidade do modelo em discriminar gestos visualmente semelhantes, bem como sua sensibilidade a variações na pose corporal, expressões faciais e configurações manuais — aspectos fundamentais na estruturação dos sinais em LIBRAS.

3.3. Extração de Keypoints

A extração de características dos vídeos foi realizada utilizando o modelo Holistic da biblioteca MediaPipe, que permite a detecção simultânea de pontos de referência na face (468), mãos (21 para cada) e corpo (33). Cada frame é convertido em um vetor de 1662 dimensões, resultante da concatenação dos pontos tridimensionais (x, y, z) e, no caso da pose, também a visibilidade. As imagens são processadas a uma taxa de 1 frame por segundo, e os vetores correspondentes são armazenados sequencialmente em arquivos .npy para posterior uso como entrada na rede.

3.4. Definição e Arquitetura do Modelo

O modelo desenvolvido é baseado em uma arquitetura do tipo LSTM (Long Short-Term Memory), uma variante de redes neurais recorrentes (RNNs) especialmente projetada para lidar com dados sequenciais e temporais. As redes LSTM possuem células que controlam o fluxo de informações ao longo do tempo, permitindo que o modelo "lembre" ou "esqueça" informações relevantes em sequências longas. Isso é possível graças a mecanismos chamados "portas" (gates), que regulam o armazenamento e a passagem de

informações, evitando o problema do desaparecimento ou explosão do gradiente, comum em RNNs tradicionais. Como resultado, LSTMs são capazes de capturar dependências temporais complexas em sequências de dados.

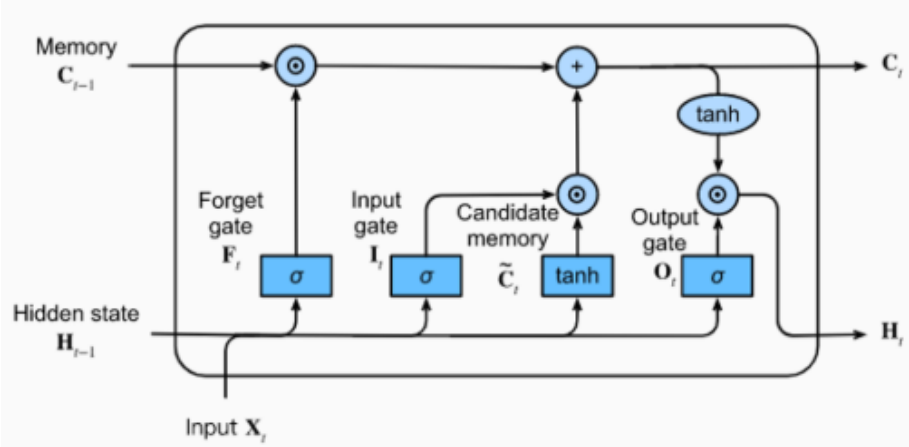


Figure 4. Arquitetura de uma rede LSTM. Naeen, J. M., & Hoque, M. M. (2023). *Architecture of LSTM with the embedding, dropout, LSTM, and dense layers*.

A escolha da arquitetura LSTM para o reconhecimento de sinais da Língua Brasileira de Sinais (LIBRAS) é particularmente adequada porque os sinais são expressos como sequências temporais de movimentos das mãos, face e corpo. Cada frame do vídeo contém informações espaciais que, em sequência, formam o gesto completo. Assim, capturar a dinâmica temporal desses gestos é fundamental para um bom desempenho no reconhecimento.

Além disso, a arquitetura inclui camadas densas com funções de ativação ReLU para extração de características não lineares, e camadas de Dropout para evitar overfitting, garantindo que o modelo generalize melhor para novos dados.

3.4.1. Configuração do Modelo

A arquitetura do modelo foi implementada com base em uma estrutura sequencial composta por camadas especializadas no processamento de dados temporais. O objetivo principal foi permitir o aprendizado de padrões espaço-temporais a partir de sequências de keypoints extraídos de vídeos da Língua Brasileira de Sinais (LIBRAS).

A entrada do modelo consiste em sequências com 10 frames, cada um representado por um vetor de 1662 atributos, correspondentes às coordenadas tridimensionais dos pontos corporais detectados. A primeira etapa do pipeline aplica uma normalização dos dados, seguida de uma camada de mascaramento, que ignora frames nulos na sequência.

A modelagem temporal é realizada por meio de uma camada LSTM composta por 512 unidades e ativação ReLU, incorporando regularização L1 com coeficiente $\lambda = 0{,}001$, a fim de reduzir o risco de sobreajuste. Em seguida, aplica-se uma camada de dropout com taxa de 40%. A camada final é densa, com 37 unidades e ativação *softmax*, refletindo o número total de classes previstas no conjunto de dados.

O modelo foi compilado com o otimizador AdamW, adotando uma taxa de aprendizado de 10^{-4} e decaimento de pesos igual a 5×10^{-3}. A função de perda utilizada foi a *categorical crossentropy*, apropriada para tarefas de classificação multiclasse com codificação one-hot. Como métricas de avaliação, foram utilizadas a acurácia padrão e a acurácia top-5, esta última permitindo analisar a presença da classe correta entre as cinco predições mais prováveis.

Table 2. Resumo das configurações da arquitetura e parâmetros do modelo.

Componente	Configuração
Entrada	Sequência com 10 frames de 1662 atributos
Normalização	Padronização dos valores de entrada
Mascaramento	Ignora frames nulos (máscara com valor zero)
LSTM	512 unidades, ativação ReLU, regularização L1 ($\lambda = 0{,}001$)
Dropout	Taxa de 40%
Camada de saída	37 unidades, ativação *softmax*
Otimizador	AdamW (lr = 10^{-4}, weight decay = 5×10^{-3})
Função de perda	Categorical crossentropy
Métricas	Acurácia e Top-5 Categorical Accuracy

3.4.2. Treinamento do Modelo

O processo de treinamento foi estruturado com base em uma divisão estratégica dos dados entre os conjuntos de treino e teste, com o objetivo de avaliar a capacidade de generalização do modelo em cenários interbase. As instâncias do articulador 2 da base V-LIBRASIL (UFPE) e da base Signbank (UFSC) foram utilizadas exclusivamente para o treinamento da rede, enquanto a base CEAD (UFV) foi reservada para testes, atuando como conjunto de validação externa.

Essa configuração foi inspirada na abordagem proposta por Sarmento & Ponti (2023), que enfatiza a importância de avaliar modelos de reconhecimento de sinais em contextos nos quais os dados de teste provêm de fontes diferentes daquelas utilizadas durante o treinamento. Tal estratégia busca refletir situações realistas de uso, em que o sistema precisa reconhecer sinais emitidos por indivíduos distintos, gravados sob diferentes condições técnicas e estilísticas.

O treinamento foi conduzido por 160 épocas, com tamanho de lote igual a 16. A função de perda utilizada foi a *categorical crossentropy*, apropriada para tarefas de classificação multiclasse com codificação one-hot. O otimizador escolhido foi o AdamW, com taxa de aprendizado de 10^{-4} e decaimento de pesos igual a 5×10^{-3}. Como métricas de avaliação, foram monitoradas a acurácia padrão e a acurácia top-5, que indica se a classe correta está entre as cinco predições com maior probabilidade.

Com o intuito de aumentar a robustez do modelo frente a pequenas variações nos dados de entrada e mitigar o risco de sobreajuste, foram aplicadas técnicas de *data augmentation* diretamente sobre os vetores de keypoints. As transformações utilizadas consistiram na injeção de ruído gaussiano com desvio padrão de $0{,}01$ e no escalonamento

aleatório com fator uniforme no intervalo $[0{,}95,\ 1{,}05]$. Cada sequência original foi replicada cinco vezes com perturbações distintas, totalizando cinco versões aumentadas por amostra original. Essa estratégia, também inspirada nas práticas descritas no estudo de referência, contribui para melhorar a capacidade do modelo em lidar com a variabilidade natural da execução dos sinais.

A etapa de teste foi realizada exclusivamente com os dados da base CEAD (UFV), que não foram utilizados em nenhum momento do treinamento. Essa escolha permitiu avaliar a capacidade do modelo de generalizar para uma fonte externa, reproduzindo um cenário realista em que os sinais são executados por indivíduos diferentes, capturados sob outras condições de gravação. Como o modelo não teve acesso prévio a esse conjunto, os resultados obtidos refletem de forma mais fiel sua performance em contextos não vistos, garantindo uma avaliação mais rigorosa da eficácia da arquitetura proposta.

3.5. Resultados e Análise Crítica

3.5.1. Desempenho do Modelo

O modelo proposto foi treinado por 160 épocas com tamanho de lote igual a 16, utilizando as bases V-LIBRASIL (Articulador 2 – UFPE) e Signbank (UFSC) como dados de treinamento, enquanto a base CEAD (UFV) foi reservada exclusivamente para teste externo. As métricas obtidas no conjunto de teste foram:

- **Acurácia (teste)**: 40,69%
- **Top-5 Accuracy (teste)**: 82,22%
- **Loss (teste)**: 3,0789

Para o conjunto de validação monitorado durante o treinamento, foram observados os seguintes resultados:

- **Acurácia (validação)**: 20,27%
- **Top-5 Accuracy (validação)**: 45,05%
- **Loss (validação)**: 5,1358

Tais resultados evidenciam a capacidade do modelo em discriminar adequadamente os sinais de LIBRAS, mesmo em um contexto multiclasse com 37 categorias distintas. A elevada taxa de acerto no Top-5 sugere que, mesmo nos casos de erro, o modelo frequentemente inclui a classe correta entre as cinco alternativas mais prováveis.

3.5.2. Evolução do Desempenho ao Longo do Treinamento

A Figura 5 apresenta a evolução das métricas de desempenho durante as 120 épocas de treinamento do modelo. No primeiro gráfico, observa-se uma rápida redução da função de perda (*loss*) no conjunto de treino, seguida de uma estabilização próxima a zero. Já a curva de validação apresenta oscilações mais acentuadas, estabilizando-se em um valor relativamente alto a partir da época 30, o que indica sinais de sobreajuste (*overfitting*).

No segundo gráfico, a acurácia de treino evolui de forma consistente, atingindo valores superiores a 95% ao final do processo. Em contraste, a acurácia de validação permanece em níveis consideravelmente inferiores, com picos abaixo de 25%. Essa divergência entre as curvas confirma a dificuldade do modelo em generalizar para dados não vistos durante o treinamento.

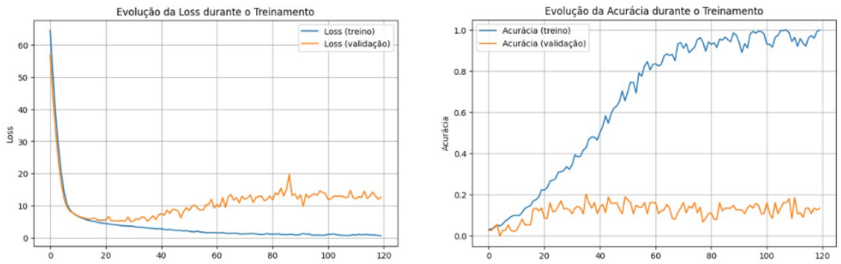


Figure 5. Evolução das métricas de loss (superior) e acurácia (inferior) ao longo das 120 épocas de treinamento.

3.5.3. Visualização de Amostras Processadas

A Figura 6 apresenta exemplos visuais de quadros anotados com pontos-chave (*keypoints*) extraídos automaticamente pelo modelo MediaPipe Holistic. Cada imagem corresponde a uma instância proveniente das diferentes bases utilizadas no experimento, permitindo visualizar a diversidade dos dados em termos de cenário, iluminação, enquadramento e morfologia dos articuladores. As amostras contemplam tanto sujeitos distintos quanto variações nas configurações corporais e faciais, elementos centrais na composição dos sinais em LIBRAS.

Observa-se que fatores como o fundo (preto, branco ou azul), a qualidade da iluminação e o posicionamento do corpo no vídeo introduzem desafios adicionais ao processo de generalização do modelo. Diferenças sutis na configuração manual, na orientação do tronco ou na expressão facial podem impactar diretamente na extração dos descritores e, por consequência, na eficácia da etapa de classificação.

Essa visualização evidencia a complexidade envolvida na tarefa de reconhecimento automático de sinais em ambientes reais. Além de ilustrar a heterogeneidade das condições de captura entre as bases, reforça-se aqui a importância da normalização geométrica e da padronização dos dados para mitigar ruídos visuais. Tais aspectos são especialmente relevantes no contexto de sinais com alto grau de similaridade gestual, onde pequenas variações podem comprometer a acurácia do modelo.

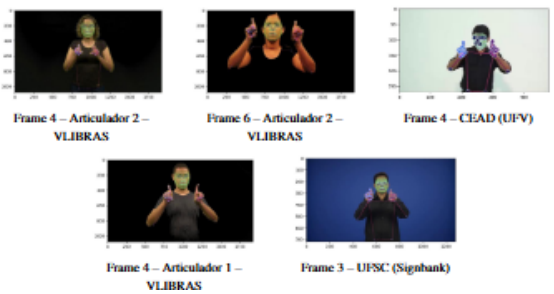


Figure 6. Amostras extraídas das bases utilizadas no experimento, anotadas com pontos-chave (*keypoints*) extraídos pelo modelo MediaPipe Holistic.

3.5.4. Análise Crítica

Apesar dos bons resultados obtidos no conjunto de treinamento, o modelo demonstrou dificuldade em generalizar para os dados de validação. A divergência acentuada entre as curvas de acurácia e perda indica sobreajuste, com desempenho inferior ao esperado nos dados não vistos. Essa limitação sugere que o modelo aprendeu padrões específicos do conjunto de treino, mas não foi capaz de capturar características mais gerais dos sinais.

A análise visual das amostras evidencia que fatores como fundo, iluminação e morfologia dos articuladores impactam significativamente o comportamento do modelo. Mesmo com a redefinição das classes e a coleta de novos exemplos, o sistema ainda se mostrou sensível à variação entre as bases. Esses aspectos reforçam a necessidade de estratégias adicionais para aumentar a robustez e melhorar o desempenho em cenários mais diversos.

4. Referências

SARMENTO, Amanda Hellen de Avellar; PONTI, Moacir Antonelli. *A Cross-Dataset Study on the Brazilian Sign Language Translation*. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 2023. Disponível em: `https://github.com/avellar-amanda/LIBRAS-Translation/`. Acesso em: 8 jun. 2025.

UNIVERSIDADE FEDERAL DE SANTA CATARINA. *SignBank – Dicionário de Língua Brasileira de Sinais*. Disponível em: `https://signbank.libras.ufsc.br/pt/`. Acesso em: 8 jun. 2025.

UNIVERSIDADE FEDERAL DE VIÇOSA. *Dicionário de Libras – Projeto Inovar Mais*. Disponível em: `https://sistemas.cead.ufv.br/capes/dicionario/`. Acesso em: 8 jun. 2025.