UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Karolayne Teixeira da Silva

SignWriting para Reconhecimento de Gestos em Língua de Sinais

Recife

2025

Karolayne Teixeira da Silva

SignWriting para Reconhecimento de Gestos em Língua de Sinais

Trabalho apresentado ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientador (a): Prof. Dr. Tsang Ing Ren

Recife

2025

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Silva, Karolayne Teixeira da.
SignWriting para reconhecimento de gestos em língua de sinais / Karolayne Teixeira da Silva. - Recife, 2025.
80f.: il.

Dissertação (Mestrado)- Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós-Graduação em Ciência da Computação, 2025.
Orientação: Tsang Ing Ren.

1. Reconhecimento de gestos; 2. SignWriting; 3. Aprendizado profundo. I. Ren, Tsang Ing. II. Título.

UFPE-Biblioteca Central

Karolayne Teixeira da Silva

“SignWriting para Reconhecimento de Gestos em Língua de Sinais”

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovado em: 25/07/2025.

BANCA EXAMINADORA

_____

Prof. Dr. George Darmiton da Cunha Cavalcanti
Centro de Informática / UFPE

_____

Prof. Dr. Alceu de Souza Brito Júnior
Programa de Pós Graduação em Informática Aplicada / PUC/PR

_____

Prof. Dr. Tsang Ing Ren
Centro de Informática / UFPE
(**orientador**)

Dedico este trabalho à minha família, que foi meu alicerce e fonte inesgotável de inspiração ao longo dessa intensa jornada de aprendizado. Ao meu pai, em especial, que, com paciência e dedicação, me escutou nos momentos de incerteza, compartilhou horas de conversas sobre soluções mesmo sem fazer ideia do que eu estava falando e ainda tirou incontáveis fotos para que eu pudesse testar minhas hipóteses. Sem o apoio e o amor incondicional de vocês, este sonho não teria se tornado realidade.

AGRADECIMENTOS

Agradeço, primeiramente, aos meus pais, Cleonice Teixeira e Edercio Rodrigues, por todo amor, dedicação e suporte incondicional, que foram fundamentais para que eu pudesse superar cada desafio desta caminhada. À minha irmãzinha, Karine Rodrigues, que, com sua alegria e energia contagiante, mesmo tão pequena, sempre esteve presente e participou de tudo de forma especial.

Aos meus amigos Thiago Buarque e Tiago Barbosa, que dedicaram horas para ouvir minhas teorias, compartilhar ideias e complementar meus testes, contribuindo significativamente para a evolução deste trabalho.

Ao meu orientador, Prof. Dr. Tsang Ing Ren, por me desafiar desde o início, incentivando meu crescimento acadêmico e possibilitando que este trabalho alcançasse o nível que apresenta hoje.

Agradeço também ao Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE), pela excelente infraestrutura fornecida.

Por fim, aos demais familiares, colegas de curso e amigos que, direta ou indiretamente, contribuíram para a conclusão desta etapa.

RESUMO

O reconhecimento automático de gestos é essencial para promover a comunicação inclusiva, sobretudo junto às comunidades Surdas. Entretanto, persistem desafios significativos em função da diversidade linguística das línguas de sinais e das limitações das abordagens convencionais, as quais tipicamente exigem grandes conjuntos de dados rotulados e apresentam baixo potencial de generalização entre diferentes idiomas, comprometendo a escalabilidade e aplicabilidade prática. Neste contexto, este trabalho propõe a utilização do SignWriting, um sistema padronizado de notação visual que codifica gestos de forma independente do idioma, como alternativa para um reconhecimento universal de gestos estáticos das mãos. A metodologia emprega o MediaPipe para extração automática de marcos anatômicos das mãos, seguida de técnicas de normalização espacial e aumento sintético de dados a fim de mitigar variabilidades individuais e ambientais. O modelo foi avaliado em 16 conjuntos de dados distintos, abrangendo 132 classes de gestos provenientes de múltiplas regiões e línguas de sinais. Os resultados obtidos indicam robustez na generalização entre línguas, corroborando o potencial do SignWriting como ferramenta unificadora. Adicionalmente, análises de sensibilidade evidenciaram a influência dos erros de detecção de marcos sobre o desempenho do classificador, apontando direções para futuras melhorias. Todo o código-fonte encontra-se disponível no repositório público: <https://github.com/karo-txs/signwriting-recognition>.

Palavras-chave: Reconhecimento de Gestos; SignWriting; Aprendizado Profundo; Aumento de Dados.

ABSTRACT

Automatic gesture recognition is essential for promoting inclusive communication, especially within Deaf communities. However, significant challenges persist due to the linguistic diversity of sign languages and the limitations of conventional approaches, which typically require large labeled datasets and have low potential for generalization across languages, compromising scalability and practical applicability. In this context, this work proposes the use of SignWriting, a standardized visual notation system that encodes gestures independently of language, as an alternative for the universal recognition of static hand gestures. The methodology employs MediaPipe for automatic extraction of hand anatomical landmarks, followed by spatial normalization and synthetic data-augmentation techniques to mitigate individual and environmental variability. The model was evaluated on 16 distinct datasets, covering 132 gesture classes from multiple regions and sign languages. The obtained results indicate robustness in cross-language generalization, corroborating the potential of SignWriting as a unifying tool. Additionally, sensitivity analyses revealed the influence of landmark-detection errors on classifier performance, pointing to directions for future improvements. All source code is available in the public repository: <https://github.com/karo-txs/signwriting-recognition>.

Keywords: Gesture Recognition; SignWriting; Deep Learning; Data Augmentation.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

As línguas de sinais são sistemas linguísticos complexos que utilizam predominantemente movimentos das mãos, expressões faciais e posturas corporais, constituindo-se como o principal meio de comunicação das comunidades Surdas (PRIEUR et al., 2020). Além de fundamentais para a expressão e interação social, essas línguas desempenham papel central na promoção da inclusão, garantindo maior acesso à educação, ao trabalho e a serviços essenciais. Entretanto, barreiras comunicacionais entre Surdos e Ouvintes ainda persistem, limitando a plena integração dessas comunidades (MANZOOR et al., 2024; SABATO; SANDRONI; MARCECA, 2023).

Com os avanços em visão computacional e aprendizado profundo, o Reconhecimento Automático de Línguas de Sinais (SLR, do inglês *Sign Language Recognition*) tornou-se uma área de pesquisa relevante, apresentando soluções para reduzir tais barreiras (ROBERT; DURAISAMY, 2023; AL-QURISHI; KHALID; SOUISSI, 2021; ALAYED, 2024; CHEOK; OMAR; JAWARD, 2017). Esses sistemas têm potencial para traduzir gestos em representações compreensíveis por máquinas, viabilizando aplicações como tradutores em tempo real, recursos educacionais interativos e ferramentas assistivas para maior autonomia dos Surdos.

Nas últimas décadas, progressos notáveis em aprendizado profundo têm impulsionado significativamente o desenvolvimento de métodos voltados ao SLR, ampliando a eficiência e precisão dessa tarefa. Dentre as técnicas mais utilizadas, destacam-se as Redes Neurais Convolucionais (CNNs, do inglês *Convolutional Neural Networks*) (POORNIMA; SRINATH, 2024; KUMAR et al., 2024; GANGWAR et al., 2024; RANGU et al., 2024; GULATI; RAJPUT; SINGH, 2024; DHANALAKSHMI et al., 2024), as Redes Neurais Recorrentes (RNNs, do inglês *Recurrent Neural Networks*) e suas variantes, como as Redes de Memória de Longo e Curto Prazo (LSTM, do inglês *Long Short-Term Memory*) (YEWARE et al., 2023; PURI et al., 2023; GANDHE et al., 2024; HUANG; CHOUVATUT, 2024), além dos *Transformers* Visuais (ViTs, do inglês *Vision Transformers*) (ALNABIH; MAGHARI, 2024; GUPTA et al., 2022; ZHANG et al., 2023), que recentemente ganharam popularidade. As CNNs destacam-se na análise de gestos estáticos devido ao seu alto desempenho na extração automática de características espaciais relevantes a partir de imagens, enquanto as RNNs e LSTMs têm sido aplicadas com sucesso em cenários dinâmicos, nos quais é fundamental a capacidade de modelar dependências temporais. Mais recentemente, os ViTs introduziram melhorias significativas ao utilizarem mecanismos de atenção, resultando em maior eficácia no processamento visual e na captura de dependências globais das imagens.

Entretanto, diversos desafios ainda limitam a generalização e escalabilidade desses sistemas. Uma limitação relevante das abordagens existentes é o foco predominante em línguas de sinais específicas, como a Língua de Sinais Americana (ASL, do inglês *American Sign Language*) (BHATT; MALIK; INDRA, 2024; NASR; KADER, 2023; ABDULLAH et al., 2023; JOURNAL, 2023), Árabe (ArSL, do inglês *Arabic Sign Language*) (ALABBAD et al., 2022; ELSHAER et al., 2024; KHATTAB et al., 2024; HASSAN; SABRI; ALI, 2024), Indiana (ISL, do inglês *Indian Sign Language*) (SIDHU et al., 2024; PASSI et al., 2024; SHIRUDE et al., 2024; SRIKANTARAO et al., 2024) e, com menor frequência a Brasileira (LIBRAS, Língua Brasileira de Sinais) (BHARTI; BALMIK; NANDY, 2023; AWAD; KOYUNCU, 2022; FURTADO; OLIVEIRA; SHIRMOHAMMADI, 2023), dificultando a criação de sistemas capazes de atender às centenas de línguas de sinais existentes ao redor do mundo.

Além disso, essas abordagens frequentemente requerem conjuntos extensos de dados rotulados, cuja obtenção é um processo oneroso e trabalhoso, limitando significativamente sua aplicabilidade prática e escalabilidade. Outro desafio crítico é a variabilidade cultural e regional das línguas de sinais, na qual gestos aparentemente idênticos podem possuir significados distintos ou até mesmo ofensivos dependendo do contexto sociocultural em que são utilizados (SINDHU et al., 2024; ABDULLAH; AMOUDI; ALGHAMDI, 2024; WAGHMARE, 2023).

A diversidade cultural entre as mais de 150 línguas de sinais existentes não se limita ao nível fonológico, mas também se manifesta no campo semântico. Isso significa que um mesmo arranjo manual pode assumir significados distintos dependendo da comunidade linguística. Por exemplo, na ASL, o gesto formado pela mão em configuração "T" (polegar entre o indicador e o médio) é utilizado para representar *bathroom/toilet*, enquanto em LIBRAS esse mesmo formato corresponde apenas à letra "T" do alfabeto manual, sem qualquer valor semântico adicional.

Outro exemplo ocorre com o gesto popularmente associado ao *I love you* na ASL: embora amplamente reconhecido como expressão afetiva no contexto norte-americano, em outras línguas de sinais, como a ArSL, é interpretado apenas como uma letra isolada, sem transmitir o mesmo significado. Tais discrepâncias evidenciam o risco de treinar classificadores baseados em dados de uma única comunidade e aplicar os resultados, sem adaptação, em diferentes contextos culturais, comprometendo a precisão e a adequação da interpretação automática.

Uma alternativa promissora para superar essas limitações técnicas é o uso de sistemas simbólicos padronizados, como o SignWriting (SUTTON, 1974a) e o Sistema de Notação de Hamburgo (HamNoSys, do inglês *Hamburg Notation System*) (PRILLWITZ et al., 1989). Tais

sistemas fornecem formas padronizadas de representação dos gestos, promovendo uma abstração simbólica consistente e interpretável tanto por humanos quanto por máquinas. Em particular, o SignWriting destaca-se devido à sua estrutura visual e universal, que permite representar claramente configurações das mãos e movimentos corporais de maneira independente do idioma utilizado. Essa característica torna o SignWriting especialmente atrativo para pesquisas em reconhecimento automático, por facilitar a generalização entre diferentes línguas de sinais. Dessa forma, essa abordagem surge como uma alternativa viável para o desenvolvimento de modelos computacionais escaláveis e generalizáveis, capazes de atender diferentes comunidades linguísticas.

Ao representar os gestos em uma forma padronizada e escrita, o SignWriting possibilita abordar o reconhecimento automático de gestos como um problema de interpretação textual. Essa perspectiva facilita o desenvolvimento de sistemas automáticos de reconhecimento de gestos mais inclusivos, escaláveis e independentes de idiomas específicos. Ademais, o uso dessa notação visual permite a aplicação direta de técnicas computacionais, como normalização espacial e aumento sintético dos dados, promovendo maior robustez perante as variabilidades individuais e contextuais comuns em ambientes reais, como diferenças no estilo dos usuários e condições ambientais diversas. Dessa forma, o SignWriting abre possibilidades tecnológicas para o desenvolvimento de sistemas práticos e globalmente acessíveis.

Nesse sentido, este trabalho propõe-se à investigação do SignWriting como notação universal para o reconhecimento de gestos estáticos das mãos, avaliando-o em 16 conjuntos de dados que totalizam 132 classes distintas, provenientes de diferentes línguas de sinais. Para alcançar esse objetivo, é introduzido um método baseado na detecção de marcos anatômicos das mãos (*landmarks*) pelo MediaPipe, os quais passam por um módulo de normalização geométrica que reduz variações de translação, rotação e escala. Essa etapa, que constitui a principal contribuição do trabalho, gera representações mais estáveis e invariantes, facilitando a tarefa de classificação mesmo em cenários com recursos limitados de dados. Em seguida, os vetores normalizados são processados por uma rede totalmente conectada (FC, do inglês *Fully Connected*) de pequena escala, que produz a distribuição de probabilidade sobre as classes de gestos e realiza o mapeamento final para os símbolos correspondentes no SignWriting.

Cabe salientar, entretanto, que a tradução completa de língua de sinais para texto ou áudio constitui um processo mais amplo, que envolve também outras etapas essenciais, tais como o reconhecimento de movimentos dinâmicos e expressões faciais, elementos estes que não são contemplados no escopo deste estudo. Assim, o reconhecimento dos gestos estáticos das mãos

configura-se como uma etapa inicial e fundamental dentro de um *pipeline* mais abrangente de tradução automática. Ademais, o êxito da metodologia proposta reforça a viabilidade de aplicações escaláveis que, baseadas em sinais visuais, promovem a inclusão socio-digital da comunidade Surda no âmago da sociedade.

1.1 OBJETIVOS

O objetivo central desta dissertação é propor, implementar e avaliar um *pipeline* de treinamento dataset-agnóstico para reconhecimento em tempo real de gestos manuais estáticos, fundamentado na representação gráfica universal SignWriting. O pipeline foi concebido para oferecer alta capacidade de generalização entre distintas línguas de sinais, reduzindo de forma significativa o esforço de adaptação a novas comunidades surdas.

Com base no objetivo principal, os seguintes objetivos específicos foram definidos para orientar o desenvolvimento deste trabalho:

- **Propor e implementar um modelo computacional universal para reconhecimento automático de gestos estáticos das mãos**: Desenvolver uma abordagem baseada no SignWriting como representação simbólica central, visando superar limitações relacionadas a barreiras linguísticas, culturais e regionais, a fim de oferecer uma solução escalável e aplicável a diferentes comunidades Surdas globalmente.
- **Integrar técnicas de normalização e aumento sintético dos dados**: Aplicar estratégias robustas para gerenciar variabilidades individuais (como tamanhos, formatos e posturas das mãos) e contextuais (condições de iluminação, ângulos de captura e ruído ambiental), visando aumentar a robustez e generalização dos modelos desenvolvidos.
- **Avaliar experimentalmente a eficácia e generalização da abordagem proposta**: Realizar testes com múltiplos conjuntos de dados representativos, abrangendo diferentes idiomas e contextos culturais, com o intuito de validar o desempenho prático e a capacidade de generalização do modelo.
- **Validar a viabilidade técnica do sistema proposto em tempo real**: Avaliar o desempenho operacional da solução proposta por meio de métricas, tais como tempo de inferência, consumo de recursos computacionais e acurácia, garantindo que o sistema possa ser empregado eficazmente em dispositivos com restrições computacionais.

1.2 ESTRUTURA DA DISSERTAÇÃO

O restante deste trabalho está organizado em cinco capítulos: o Capítulo 2 apresenta os fundamentos teóricos necessários para a compreensão do reconhecimento de gestos e do sistema SignWriting; o Capítulo 3 revisa trabalhos relacionados, abordando sistemas de escrita visual e métodos modernos de reconhecimento de gestos; o Capítulo 4 detalha o modelo proposto, desde a extração de *landmarks* com Mediapipe até o treinamento do modelo; o Capítulo 5 expõe os resultados experimentais, validando a robustez e aplicabilidade do sistema; e o Capítulo 6 conclui a pesquisa, destacando suas contribuições e propondo direções futuras.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 NOTAÇÕES VISUAIS PARA LÍNGUAS DE SINAIS

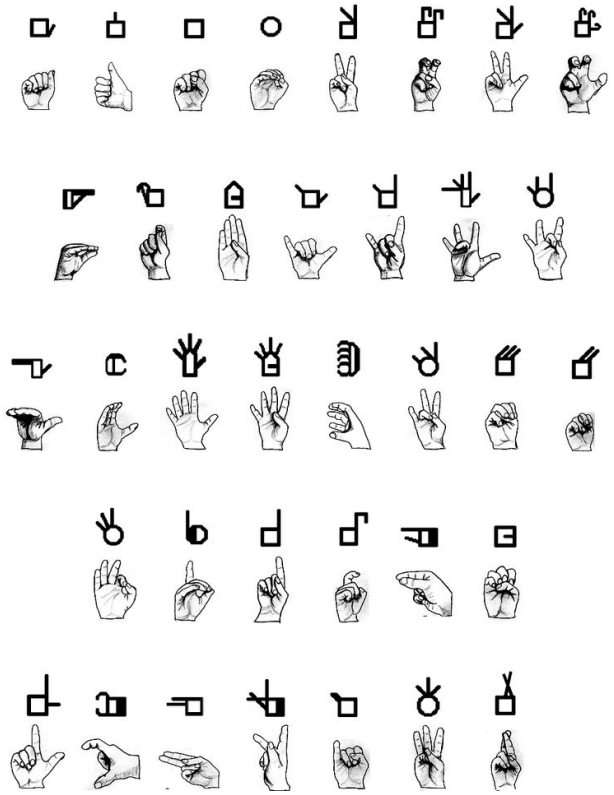
A comunicação por meio das línguas de sinais envolve uma combinação complexa de elementos visuais, incluindo configurações de mão, movimentos, expressões faciais e posturas corporais (CHEOK; OMAR; JAWARD, 2017; NEIVA; ZANCHETTIN, 2018). Para codificar tais particularidades linguísticas de maneira padronizada e estruturada, diversos sistemas formais de notação visual têm sido desenvolvidos ao longo das últimas décadas. Entre esses sistemas destacam-se o SignWriting e o HamNoSys.

2.1.1 SignWriting

O SignWriting é um sistema visual padronizado, criado por Valerie Sutton em 1974, concebido para representar de maneira estruturada e independente do idioma os principais elementos linguísticos das línguas de sinais, tais como a configuração e orientação das mãos, trajetórias de movimento, localização espacial relativa ao corpo e expressões faciais (SUTTON, 1974a). A Figura 1 ilustra alguns exemplos representativos dos símbolos utilizados no SignWriting para descrever configurações das mãos comumente encontradas nas línguas de sinais.
Uma das características mais notáveis do SignWriting é sua alta acessibilidade gráfica, permitindo que sinais sejam lidos e escritos mesmo por usuários sem conhecimento prévio de linguística formal. Criado por Valerie Sutton como uma forma visual intuitiva para documentação das línguas de sinais, esse sistema facilita não apenas a comunicação, mas também abre possibilidades para aplicações tecnológicas.

No contexto específico do reconhecimento automático de línguas de sinais, o SignWriting surge como uma alternativa poderosa para lidar com a diversidade linguística e cultural. Sua estrutura simbólica é projetada para ser universal e independente do idioma (STIEHL et al., 2015), o que favorece sua aplicação em cenários multiculturais. No entanto, o mapeamento automático de sinais representados em vídeos ou marcos anatômicos das mãos (*landmarks*) para símbolos gráficos do SignWriting constitui um desafio computacional significativo. Para enfrentar tais desafios, é necessário recorrer a técnicas de visão computacional e aprendizado de máquina, garantindo representações robustas e precisas dos gestos, mesmo em condições variáveis de captura e interpretação.

Figura 1 – Exemplos de símbolos do SignWriting que descrevem diferentes configurações de mão em línguas de sinais.



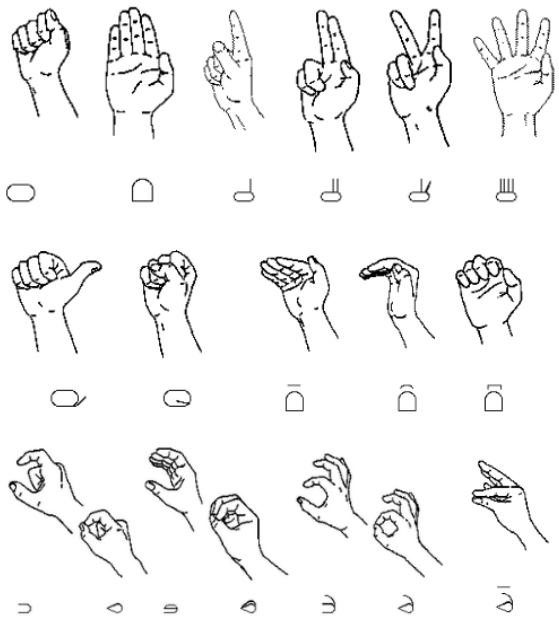
Fonte: Wikipedia, 2007. Disponível em: <https://en.wikipedia.org/wiki/File:Handshape_equiv2.png>.

2.1.2 Hamburg Notation System (HamNoSys)

O HamNoSys foi desenvolvido no final da década de 1980 pelo Instituto de Linguística da Universidade de Hamburgo, visando fornecer uma representação fonológica detalhada e padronizada para as línguas de sinais (PRILLWITZ et al., 1989). O sistema baseia-se em um conjunto detalhado de símbolos gráficos que descrevem precisamente elementos fonológicos das línguas de sinais, tais como configuração das mãos, localização espacial relativa ao corpo, orientação das mãos e trajetórias dos movimentos realizados, conforme ilustrado na Figura 2.

Embora o HamNoSys proporcione um maior grau de detalhamento fonológico quando comparado ao SignWriting, o sistema apresenta desafios significativos que restringem sua adoção ampla tanto em contextos linguísticos quanto tecnológicos. Estudos como o de Ferlin et al. (FERLIN; MAJCHROWSKA; NALEPA, 2024) destacam inconsistências frequentes na rotulagem dos símbolos e uma curva de aprendizado acentuada, fatores que representam barreiras importantes para usuários e desenvolvedores. Adicionalmente, a existência de variações nas formas gráficas dos símbolos do HamNoSys torna desafiadora a padronização consistente de conjuntos

Figura 2 – Exemplos do conjunto de símbolos do HamNoSys, representando configurações de mão.



Fonte: Adaptado de Universidade de Hamburgo, s.d. Disponível em: <https://www.sign-lang.uni-hamburg.de/projekte/hamnosys/hamnosyserklaerungen/englisch/contents.html>.

de dados, aumentando consideravelmente a complexidade envolvida na sua aplicação computacional e na criação de modelos automáticos robustos (FERLIN; MAJCHROWSKA; NALEPA, 2024).

2.1.3 Comparação e Relevância para Aplicações Computacionais

O SignWriting e o HamNoSys possuem características distintas que podem ser consideradas complementares, diferindo principalmente em termos de acessibilidade gráfica, complexidade técnica e detalhamento fonológico. Enquanto o SignWriting é reconhecido por sua alta intuitividade gráfica e facilidade de aprendizado, o HamNoSys destaca-se sobretudo pela precisão técnica na descrição detalhada dos aspectos fonológicos das línguas de sinais, sendo particularmente valorizado em estudos linguísticos formais (PRILLWITZ et al., 1989).

Na Figura 3, apresentamos um exemplo comparativo entre as duas notações: em (a), observa-se a composição de um sinal no SignWriting, cuja representação é visualmente clara e diretamente associada à configuração manual; em (b), a mesma informação é registrada em HamNoSys, cuja estrutura simbólica, embora tecnicamente mais precisa, é menos intuitiva para usuários não especialistas. Essa diferença ilustra de maneira prática como a notação SignWriting tende a ser mais acessível e compreensível em cenários de aplicação computacional.

No contexto específico deste trabalho, optou-se pelo uso do SignWriting devido à sua simplicidade gráfica, maior acessibilidade para usuários sem treinamento linguístico especializado e sua proposta de universalidade simbólica, aspectos que favorecem o desenvolvimento de sistemas computacionais escaláveis e com maior potencial de generalização (BIANCHINI; BORGIA; MARSICO, 2012; BOUZID; JEMNI, 2013; STIEHL et al., 2015; SEVILLA; ESTEBAN; LAHOZ-BENGOECHEA, 2023). Essa escolha facilita o processo de mapeamento automático dos dados capturados, tais como os marcos anatômicos das mãos, para representações simbólicas estruturadas, promovendo maior consistência, robustez técnica e aplicabilidade em contextos multiculturais.

Figura 3 – Comparação visual entre os sistemas SignWriting e HamNoSys.

(a) SignWriting (b) HamNoSys

Fonte: A autora (2025)

2.2 RECONHECIMENTO DE GESTOS

O reconhecimento automático de gestos consiste em um conjunto de métodos computacionais destinados à interpretação e identificação de movimentos e configurações corporais, com destaque especial para movimentos das mãos, braços, posturas e expressões faciais (GUPTA et al., 2022; SAHOO et al., 2022). A capacidade de sistemas computacionais reconhecerem automaticamente gestos possui uma ampla gama de aplicações práticas, destacando-se a Interação Humano-Computador (IHC), a acessibilidade tecnológica voltada para comunidades Surdas e aplicações em áreas como entretenimento digital, educação inclusiva e telemedicina (ZHANG et al., 2023). Nesta subseção, são discutidos os fundamentos conceituais e técnicos relacionados à natureza dos gestos, suas principais formas de representação simbólica e as categorias

metodológicas mais utilizadas no reconhecimento automático de gestos.

2.2.1 Gestos Estáticos e Dinâmicos

Uma das distinções fundamentais no reconhecimento automático de gestos refere-se à classificação entre gestos estáticos e dinâmicos. Os gestos estáticos são tipicamente definidos por configurações espaciais corporais, especialmente relacionadas à posição e forma das mãos, enquanto os gestos dinâmicos envolvem variações espaciais e temporais contínuas, demandando técnicas específicas para capturar dependências sequenciais nos dados ao longo do tempo (GüLER; YüCEDAğ, 2021).

Gestos estáticos são comumente tratados como problemas de classificação estática, usando abordagens baseadas em CNNs, ViTs ou Redes Totalmente Conectadas (FC, do inglês *Fully Connected*), especialmente adequadas quando os dados já estão estruturados em formato vetorial. Em contraste, os gestos dinâmicos requerem métodos capazes de lidar explicitamente com dependências sequenciais e temporais, sendo amplamente empregadas RNNs e suas variantes especializadas, como as LSTMs (GüLER; YüCEDAğ, 2021).

2.2.2 Aquisição e Representação dos Dados

Diversas técnicas podem ser empregadas para a aquisição e representação dos dados utilizados no reconhecimento automático de gestos. Entre as abordagens mais comuns destacam-se:

- **Câmeras RGB (2D)**: amplamente utilizadas devido à simplicidade e baixo custo, capturam imagens bidimensionais que são, entretanto, suscetíveis a variações ambientais significativas, tais como iluminação irregular, oclusões parciais e variações nos ângulos de captura (CHEOK; OMAR; JAWARD, 2017).

- **Câmeras de profundidade (3D)**: fornecem informações tridimensionais detalhadas, permitindo uma captura espacial mais precisa e reduzindo problemas associados às variações ambientais, embora ainda possam sofrer com oclusões.

- **Sensores inerciais e dispositivos vestíveis**: incluem sensores especializados, como luvas equipadas com acelerômetros e giroscópios, permitindo capturar detalhadamente movimentos tridimensionais. Apesar de fornecerem dados altamente precisos, esses dis-

positivos têm escalabilidade limitada por exigirem hardware dedicado e específico para sua implementação (MOHANDES; DERICHE; LIU, 2014).

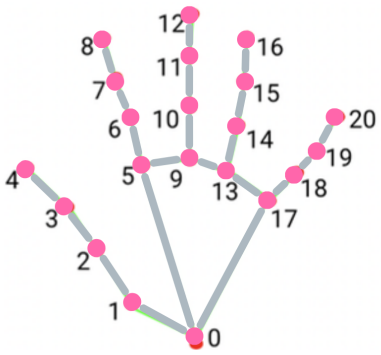
Neste estudo adota-se exclusivamente a modalidade de imagens RGB capturadas por câmeras convencionais, tanto pela ampla disponibilidade desse tipo de sensor quanto pela facilidade de integração em ambientes reais, aspectos essenciais para a reprodutibilidade e escalabilidade da proposta. As demais técnicas são discutidas apenas para contextualização e não fazem parte do escopo experimental desta pesquisa.

2.3 EXTRAÇÃO DE MARCOS (*LANDMARKS*)

Uma etapa essencial no reconhecimento automático de gestos consiste na detecção, extração e normalização de marcos anatômicos (*landmarks*), especialmente aqueles relacionados às mãos. Essa fase envolve a identificação precisa e padronizada de pontos-chave, como articulações, pontas dos dedos e posição do pulso, resultando em uma representação abstrata e compacta dos gestos realizados. Essa representação é significativamente menos suscetível a ruídos provenientes de fatores como diferentes tonalidades de pele, condições variadas de iluminação ou complexidade dos fundos das imagens capturadas.

A detecção automática desses marcos anatômicos pode ser realizada por meio de *frameworks* de Visão Computacional baseadas em aprendizado profundo, como o *MediaPipe Hand Landmark Detector* (GOOGLE, 2020), que identifica 21 marcos tridimensionais em cada mão, conforme ilustrado na Figura 4. A conversão dos dados brutos de imagens para representações vetoriais baseadas nesses marcos simplifica significativamente a tarefa computacional de reconhecimento, ao focar exclusivamente em características geométricas essenciais das mãos, reduzindo assim a dimensionalidade e complexidade dos dados utilizados pelos modelos.

Figura 4 – Marcos da mão direita conforme definidos pelo MediaPipe; para a mão esquerda, basta considerar o reflexo horizontal desses pontos.



0. Wrist
1. Thumb CMC
2. Thumb MCP
3. Thumb IP
4. Thumb TIP
5. Index Finger MCP
6. Index Funger PIP
7. Index Funger DIP
8. Index Funger TIP
9. Middle Finger MCP
10. Middle Finger PIP
11. Middle Finger DIP
12. Middle Finger TIP
13. Ring Finger MCP
14. Ring Finger PIP
15. Ring Finger DIP
16. Ring Finger TIP
17. Pinky MCP
18. Pinky PIP
19. Pinky DIP
20. Pinky TIP

Fonte: A autora (2025)

O uso de representações baseadas em marcos anatômicos apresenta diversas vantagens práticas e metodológicas no contexto do reconhecimento automático de gestos:

- **Redução significativa da dimensionalidade**: Em vez de utilizar imagens completas como entrada direta para os modelos, são processados apenas conjuntos limitados de pontos-chave (por exemplo, 21 marcos no caso do MediaPipe), diminuindo consideravelmente a complexidade computacional e permitindo maior eficiência tanto no treinamento quanto na inferência dos modelos (GUPTA et al., 2022).

- **Menor sensibilidade a variações visuais e ambientais**: Representações baseadas em marcos anatômicos são menos susceptíveis a interferências provenientes de fatores externos, como diferentes tonalidades de pele, roupas, iluminação ou complexidade dos fundos, já que o foco principal reside nas coordenadas geométricas e estruturais dos gestos.

- **Representação espacial tridimensional**: Algumas ferramentas, como o MediaPipe, fornecem coordenadas em três dimensões (x, y, z), permitindo análises mais completas da estrutura espacial das mãos. Essa representação tridimensional auxilia significativamente na redução de ambiguidades causadas por ângulos desfavoráveis e facilita a captura precisa de movimentos complexos.

Apesar dessas vantagens claras, é importante ressaltar que a eficácia das representações baseadas em marcos anatômicos depende diretamente da precisão do método utilizado para a detecção desses pontos-chave. Erros de detecção, tais como a presença de *outliers* (pontos detectados fora da posição esperada), ou imprecisões causadas por condições ambientais adversas

podem impactar negativamente a qualidade dos dados, afetando diretamente o desempenho dos modelos de reconhecimento automático subsequentes.

3 TRABALHOS RELACIONADOS

Nas duas últimas décadas, a investigação em reconhecimento automático de línguas de sinais deslocou-se de classificadores baseados em atributos manuais para arquiteturas de aprendizagem profunda que convertem sequências gestuais diretamente em glossas ou sentenças. Embora o ganho de desempenho em *datasets* de ASL, ArSL, ISL e LIBRAS seja expressivo, sobretudo após combinações CNN, LSTM e *Transformers* multimodais, o panorama permanece, em grande medida, monolíngue. Cada comunidade requer extensos ciclos de anotação e *fine-tuning*, o que eleva os custos de transferência para novos dialetos. Tal limitação renova o interesse por sistemas de escrita gestual potencialmente universais, entre os quais o SignWriting se destaca, graças à representação icônica independente de língua (SUTTON, 1974b).

3.1 RECONHECIMENTO E TRADUÇÃO DE LÍNGUAS DE SINAIS

Nas aplicações de aprendizado profundo ao reconhecimento de línguas de sinais, redes convolucionais atuaram como principais extratoras de características visuais, em cenários tipicamente isolados, restritos a gestos estáticos ou sequências curtas. Trabalhos que fazem uso de CNNs (POORNIMA; SRINATH, 2024; KUMAR et al., 2024; GANGWAR et al., 2024; RANGU et al., 2024; GULATI; RAJPUT; SINGH, 2024; DHANALAKSHMI et al., 2024; GUPTA et al., 2022) e versões leves de YOLO (NAVIN et al., 2025; BURIBAYEV et al., 2025; ALSHARIF et al., 2025) ilustram essa fase, alcançando acurácia acima de 95% em alfabetos de ASL, Bangla (BdSL, do inglês *Bangla Sign Language*) e ISL.

Com a popularização dos ViTs e de bibliotecas de detecção de pontos-chave, como MediaPipe e OpenPose, surgiram arquiteturas híbridas que combinam CNNs e ViTs, além de soluções que integram descritores de pose de mãos a fluxos de pixels (DAMDOO; KUMAR, 2025; MAIA; LOPES; DAVID, 2025; MARQUEZ et al., 2025; RODRIGUEZ et al., 2025). Apesar dos ganhos recentes em alcance e precisão, a maioria dos estudos continua a utilizar *datasets* monolíngues, o que preserva a dificuldade de generalização intralinguística que será examinada nas seções seguintes.

Entre as abordagens que mantêm a CNN como núcleo da extração visual, sobressai o trabalho de (GUPTA et al., 2022), que combina uma CNN com um ViT. O método foi avaliado nos conjuntos *NUS Hand Posture, Sign Language Digits* (Turquia) (PISHARADY; VADAKKE-

PAT; POH, 2014) e em um subconjunto alfabético de ASL, alcançando acurácia entre 90% e 99%. Ainda assim, o experimento permanece limitado a gestos estáticos de dígitos e letras provenientes dessas línguas de maior alcance, de modo que o viés monolíngue anteriormente mencionado continua vigente.

No contexto da tradução contínua de sinais em texto, os autores em (MAIA; LOPES; DAVID, 2025) combinam a detecção de pontos-chave corporais do MediaPipe com um pipeline *Transformer duplo*, segmentado em *Sign2Gloss* e *Gloss2Text*. A primeira etapa utiliza CTC-Loss, enquanto a segunda realiza ajuste fino do modelo BART. Avaliado no corpus PHOENIX14T, o sistema preservou a qualidade mesmo após forte redução de dimensionalidade. No conjunto How2Sign, entretanto, o desempenho caiu de forma acentuada, possivelmente devido à ausência de anotações de glossas, evidenciando a vulnerabilidade de métodos que dependem de glossários alinhados e de bases específicas de ASL-Alemão.

A Tabela 1 sintetiza esse panorama, comparando diferentes trabalhos recentes em reconhecimento de gestos estáticos de mãos quanto à abordagem utilizada, quantidade de conjuntos de dados, número de classes e línguas de sinais avaliadas. Nota-se que, na maioria dos casos apresentados, os experimentos permanecem restritos a poucas bases (em média até três) e a duas ou três línguas de sinais, o que reforça a dificuldade de generalização interlinguística. Em contraste, este estudo expande a análise para 16 conjuntos heterogêneos, cobrindo 132 classes distribuídas em oito línguas de sinais distintas.

Tabela 1 – Comparação com estudos da literatura sobre Reconhecimento de Poses de Mão. A tabela apresenta a abordagem utilizada, número de conjuntos de dados de teste, quantidade de classes e as línguas de sinais avaliadas. As línguas de sinais suportadas pelos conjuntos de dados são: ASL (*American Sign Language*, TSL (*Turkish Sign Language*), ISL (*Indian Sign Language*, ArSL (*Arabic Sign Language*), BdSL (*Bengali Sign Language*), LSA (*Argentine Sign Language*), DGS (*Deutsche Gebärdensprache*), PSL (*Pakistan Sign Language*) e LIBRAS (Língua Brasileira de Sinais).

Estudo	**Abordagem**	**Conjuntos de dados de teste**	**Classes**	**Línguas de Sinais**
(GUPTA et al., 2022)	CNN	3	20	ASL, TSL
(MENON; SRUTHI; LIJIYA, 2022)	CNN	1	26	ISL
(ALAMRI et al., 2024)	YOLOv8	2	32	ArSL
(SURJO et al., 2023)	VGG16	1	37	BdSL
(RONCHETTI et al., 2023)	DenseNet	3	71	LSA, DGS
(FALLAH et al., 2024)	FC-NN	3	89	ASL, ISL, BdSL
Este estudo	**Mediapipe + FC (SignWriting)**	**16**	**132**	**ASL, ISL, LSA, ArSL, BdSL, DGS, PSL e LIBRAS**

Fonte: A autora (2025).

3.2 ABORDAGENS MULTILÍNGUA

A fragmentação inerente às mais de 150 línguas de sinais existentes impõe um desafio singular aos sistemas de reconhecimento e tradução: modelos treinados em um único idioma apresentam drástica perda de desempenho quando expostos a gestos provenientes de outras comunidades linguísticas. Por essa razão, cresce o interesse em arquiteturas multilínguas capazes de compartilhar parâmetros entre diferentes conjuntos de dados e, assim, reduzir o custo de adição de novas línguas.

Entre as propostas mais robustas nessa linha, o GmTC (*Graph and General Two-Stream Network*) combina uma *Graph Convolutional Networks*, encarregado de codificar relações espaciais finas entre superpixels, com um *multi-head attention* voltado para capturar dependências de longo alcance (MIAH et al., 2024). Avaliado em cinco corpora de origens culturais distintas (coreano, americano, japonês, entre outros), o modelo alcançou média de acurácia superior a 98% sem exigir ajustes significativos na fase de pré-processamento.

Seguindo a proposta de vocabulário múltiplo, o framework OpenHands adota um protocolo baseado em poses extraídas pelo MediaPipe e disponibiliza checkpoints pré-treinados para seis línguas de sinais: ASL, Argentina (LSA, do Espanhol *Lengua de Señas Argentina*), Chinesa (CSL, do inglês *Chinese Sign Language*), Grega (GSL, do inglês *Greek Sign Language*), ISL e Turca (TSL, do inglês *Turkish Sign Language*) (SELVARAJ et al., 2022). O principal diferencial é um pré-treino auto-supervisionado sobre mais de um milhão de quadros não anotados de Indian Sign Language, cujo conhecimento se transfere para idiomas de baixo recurso e pode reduzir em até 40% a necessidade de dados rotulados.

Investigações recentes têm ampliado o espectro metodológico. O SB-SLR adota um fluxo exclusivamente esquelético, no qual quadros-pivô são identificados antes do processamento por uma CNN temporal 2-D; essa configuração produz ganhos consistentes em cenários de desequilíbrio de classe e variação de signatários, inclusive para línguas de sinais pouco estudadas, como a cazaque (RENJITH; SURESH; RASHMI, 2025). Estratégias híbridas de detecção de língua seguida de reconhecimento obtêm acurácia superior a 98% em dois idiomas (NURNOBY; EL-ALFY, 2023). Abordagens bilíngues fundamentadas no YOLOv11, por sua vez, registram mAP acima de 99% nos alfabetos de BdSL e ASL (NAVIN et al., 2025).

No campo da tradução, iniciativas como o AfriSign, que emprega *Transformers* multilíngues para seis línguas africanas (TAKYI et al., 2025), e abordagens gloss-free, a exemplo do Sign2GPT-Next, que integra um ViT Dino-V2 a um GPT multilinguístico (BABISHA et al.,

2024), indicam ser possível reduzir a dependência da camada de glossas mesmo em cenários de recursos limitados.

Apesar dos avanços, o panorama atual continua dependente de dados amplamente anotados para cada língua. Tanto a calibração de detectores de pontos-chave, cujos erros se propagam aos modelos, quanto o ajuste dos módulos de pré-processamento linguístico na etapa de tradução ainda requerem supervisão específica, fazendo o custo de expansão crescer à medida que novas comunidades linguísticas são incluídas. Essa limitação tem estimulado a busca por representações independentes de idioma que atuem como pivôs semântico-gráficos entre diferentes línguas de sinais (FINK et al., 2023; YAZDANI; GENABITH; ESPAñA-BONET, 2025).

3.3 SIGNWRITING

O SignWriting é um método que descreve configurações de mão, trajetórias, expressões faciais e orientação corporal por meio de símbolos icônicos dispostos em duas dimensões. Por ser independente de idioma, permitindo registrar qualquer sinal sem recorrer a glossas verbais, o SignWriting é um candidato natural a atuar como pivô em aplicações computacionais multilíngues. As pesquisas sobre o SignWriting têm avançado em três frentes: (i) reconhecimento automático, (ii) tradução e síntese visual e (iii) ferramentas educacionais e de acessibilidade.

No eixo de reconhecimento, os primeiros protótipos dedicaram-se à classificação de símbolos isolados. Liu et al. (2010) (LIU et al., 2010) apresentaram um sistema de interação humano–computador capaz de reconhecer determinadas trajetórias manuais e mapeá-las para símbolos do SignWriting. Avanços subsequentes exploraram CNNs para identificar conjuntos de pictogramas: o estudo de (STIEHL et al., 2015) obteve 94,4% de acerto em 7994 amostras distribuídas em 103 classes, enquanto (SEVILLA; ESTEBAN; LAHOZ-BENGOECHEA, 2023) combinou redes neurais e regras especialistas para modelar a natureza composicional dos sinais, registrando ganho relativo de 17% sobre uma abordagem baseada apenas em aprendizado profundo. Cabe ressaltar que ambas as linhas de investigação se concentram nos desenhos dos símbolos, e não no reconhecimento dos sinais a partir de imagens ou vídeos de signatários.

Um contraponto relevante às abordagens que operam exclusivamente sobre pictogramas estáticos é o *Deep Hand*, de Koller et al. (2016) (KOLLER; NEY; BOWDEN, 2016), que utiliza os códigos do SignWriting para gerar rótulos fracos dos quadros de vídeo. O trabalho introduz uma CNN pré-treinada, refinada com mais de 1 milhão de quadros rotulados automaticamente (abrangendo 60 configurações de mão distintas) e alcança 62,8% de acerto top-1. Sua principal

contribuição é mostrar que rótulos imperfeitos derivados do SignWriting podem sustentar treinamento em larga escala e ainda generalizar entre diferentes corpora e signatários. Nesse sentido, o estudo de Koller (2016) pode ser visto como precursor da ideia central explorada nesta dissertação: o uso do SignWriting como uma forma de anotação simbólica intermediária capaz de reduzir a dependência de anotações manuais extensivas e tornar viável o aprendizado em cenários de baixo recurso.

No domínio da tradução entre línguas faladas e sinais codificados em SignWriting, (JIANG et al., 2022) mostraram que técnicas clássicas de aprendizado de máquina podem ser transferidas com êxito quando o corpus de origem (SignBank) utiliza o SignWriting como representação intermediária, alcançando pontuação superior a 30 de BLEU (*Bilingual Evaluation Understudy*) para o par ASL para inglês. Mais recentemente, (FREITAS et al., 2023) empregou codificações formais em SignWriting para treinar modelos de representação latente, obtendo 81% de acurácia em uma tarefa de classificação com apenas 889 amostras, o que reforça o potencial de pesquisas em cenários de baixo recurso.

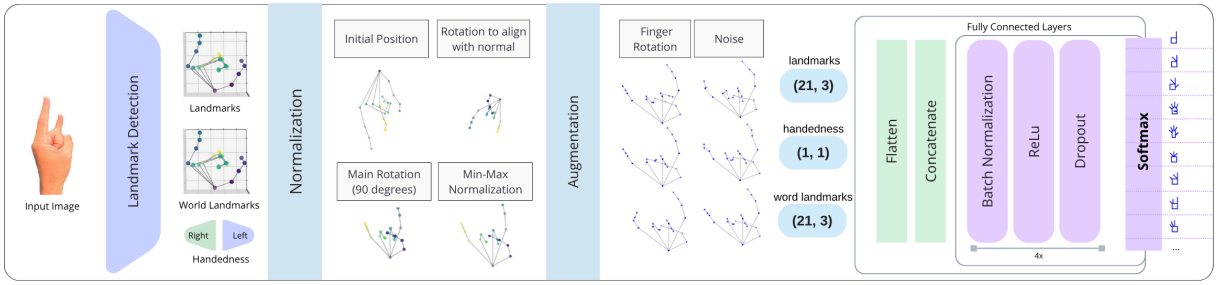
No domínio da síntese de movimentos e ambientes imersivos, diversos autores exploram SignWriting como camada de entrada para gerar animações fidedignas. (BOUZID et al., 2012) propuseram a conversão dos sinais para um avatar 3D, enquanto projetos como tuniSigner e SiGML-to-VR (BOUZID; JEMNI, 2014; WOLFF; ANDERSON; BANIć, 2024) exibem traduções em tempo real de textos ou discursos para ASL dentro de ambientes de realidade virtual. Essa linha de investigação foi estendida à acessibilidade televisiva, com a proposta de encapsular SignWriting em legendas IMSC1 para a futura TV 3.0 brasileira (LOBEIRO; VAZ; ALVES, 2022).

Em síntese, a literatura revela o papel estratégico do SignWriting como representação universal: ele viabiliza tradução, indexação textual, animação de avatares e ensino formal sem depender de glossas verbais. Todavia, a maioria dos trabalhos concentra-se em tarefas pontuais (símbolos isolados ou geração de animações) e raramente avalia a generalização interlinguística em larga escala. Estudos como (KOLLER; NEY; BOWDEN, 2016; JIANG et al., 2022; FREITAS et al., 2023) tipicamente restringem suas análises a duas ou três línguas de sinais e a menos de quatro bases de dados, o que limita a avaliação da robustez intercultural dos métodos. A presente dissertação avança nesse cenário ao explorar a classificação automática de gestos estáticos das mãos em 16 bases heterogêneas (132 classes, múltiplas línguas), evidenciando o potencial do SignWriting para unificar *pipelines* de reconhecimento e tradução em contextos multilíngues.

4 MODELO PROPOSTO

Este trabalho propõe uma metodologia para o desenvolvimento e validação experimental de um sistema automático de reconhecimento de gestos estáticos das mãos utilizando o SignWriting como representação intermediária padronizada. A escolha dessa representação visa explicitamente aumentar a generalização linguística e escalabilidade dos modelos desenvolvidos, buscando mitigar limitações técnicas frequentemente encontradas em sistemas convencionais de reconhecimento automático de línguas de sinais, como diversidade linguística, variabilidade cultural e necessidade constante de retreinamento.

Figura 5 – Fluxograma da metodologia proposta que compreende quatro etapas principais: extração automática dos marcos anatômicos (*landmarks*) das mãos, normalização espacial dos dados obtidos, geração de dados sintéticos utilizando técnicas de aumento (*data augmentation*) e a arquitetura para reconhecimento automático dos gestos estáticos das mãos.

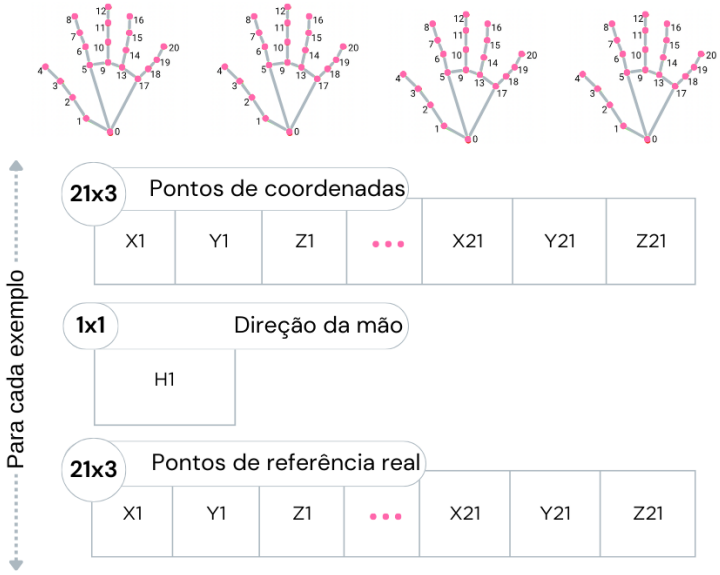


Fonte: A autora (2025)

A metodologia proposta, ilustrada na Figura 5, começa com a detecção dos marcos da mão pelo MediaPipe, resultando em três tensores independentes, como exemplificado na Figura 6:

- (a) 21 marcos em coordenadas de imagem $(21, 3)$: (x, y) normalizados ao plano da câmera e z como profundidade relativa;
- (b) *handness* $(1, 1)$: escalar que indica se a mão detectada é direita (1) ou esquerda (0);
- (c) 21 marcos em coordenadas de mundo $(21, 3)$: posições (x, y, z) no espaço tridimensional real, em escala métrica e com origem no centro da câmera.

Figura 6 – Exemplo dos tensores resultantes da detecção de marcos da mão pelo MediaPipe.



Fonte: A autora (2025)

Os tensores (a) e (c) passam, em paralelo, por um módulo de normalização que reduz variações de translação, rotação e escala. Essa etapa constitui a principal contribuição deste trabalho: ao aplicar transformações geométricas sobre os *landmarks* do MediaPipe, obtêm-se representações invariantes às diferenças de posição e orientação da mão, o que gera amostras mais estáveis e consistentes para o classificador. Como resultado, o modelo final demanda menos parâmetros, pode ser treinado com quantidades menores de dados rotulados e ainda assim mantém desempenho competitivo em múltiplas línguas de sinais.

Em seguida, cada tensor normalizado é linearizado (transformado em vetor unidimensional) e todos são concatenados, formando um único vetor de atributos. Esse vetor alimenta uma rede totalmente conectada (FC) composta por quatro blocos idênticos, cada qual organizado na ordem *Batch Normalization*, *Rectified Linear Unit* (ReLU) e Dropout. Por fim, uma camada densa com *Softmax* produz a distribuição de probabilidade sobre as classes de gestos estáticos, cujo rótulo previsto é mapeado para o símbolo correspondente no SignWriting.

4.1 EXTRAÇÃO AUTOMÁTICA DE MARCOS ANATÔMICOS COM MEDIAPIPE

A extração automática dos marcos anatômicos das mãos foi realizada utilizando a biblioteca *MediaPipe Hand Landmark Detector*[1], uma ferramenta amplamente reconhecida por sua capacidade de realizar detecções rápidas e precisas das mãos em tempo real. O Mediapipe

[1] <https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker>

fornece um conjunto fixo de 21 marcos anatômicos (*landmarks*) tridimensionais por mão, totalizando 63 valores por mão (cada ponto contendo coordenadas espaciais (x, y, z), além de fornecer informações adicionais sobre a probabilidade de lateralidade (direita ou esquerda).

4.2 NORMALIZAÇÃO DE DADOS

A normalização espacial dos dados extraídos é uma etapa crucial para o reconhecimento automático de gestos, visto que variações nas posições, orientações e escalas das mãos capturadas podem introduzir ruídos e inconsistências significativas no processo de treinamento dos modelos. Para garantir maior robustez e generalização das representações utilizadas, é realizado um procedimento de normalização espacial dos marcos anatômicos (*landmarks*), assegurando representações invariantes quanto à escala, rotação e posição espacial das mãos.

4.2.1 Cálculo do Vetor Normal da Palma da Mão

Os 21 *landmarks* 3-D fornecidos pelo MediaPipe são armazenados num tensor $\mathbf{P} \in R^{21\times3}$, em que cada linha $\mathbf{p}_i = (x_i, y_i, z_i)$ representa um ponto anatômico.

Para estimar a orientação global da mão, usamos apenas três pontos (enumerados conforme a Figura 4):

- $\mathbf{p}_0$ – pulso
- $\mathbf{p}_5$ – base do dedo
- $\mathbf{p}_{17}$ – base do dedo mínimo

Dois vetores são então formados a partir do pulso: $\mathbf{v}_1 = \mathbf{p}_{17} - \mathbf{p}_0$ e $\mathbf{v}_2 = \mathbf{p}_5 - \mathbf{p}_0$.

O produto vetorial desses vetores gera um vetor perpendicular ao plano da palma; após normalização obtemos

$$\vec{n} = \frac{\mathbf{v}_1 \times \mathbf{v}_2}{\|\mathbf{v}_1 \times \mathbf{v}_2\|}, \tag{4.1}$$

onde $\vec{n}$ tem módulo 1 e aponta para fora da palma.

4.2.2 Alinhamento dos *Landmarks* ao Plano da Palma

Depois de calcular o vetor normal $\vec{n}$, os *landmarks* são rotacionados para um sistema de coordenadas local em que

- O eixo Z aponta na direção de $\vec{n}$ (perpendicular à palma);
- O eixo X está contido na palma;
- O ponto de rotação é o pulso, que passa a ser a origem.

Esse alinhamento elimina diferenças de orientação entre mãos filmadas de ângulos distintos, reduzindo a variabilidade dos dados. É construída uma base ortonormal $\{\vec{x}, \vec{y}, \vec{z}\}$:

- $\vec{z} = \vec{n}$
- $\vec{y} = (1, 0, 0) \times \vec{z}$
- $\vec{x} = \vec{z} \times \vec{y}$

Empilhando esses vetores linha-a-linha, forma-se a matriz de rotação $R = \begin{bmatrix} \vec{x}^\top \\ \vec{y}^\top \\ \vec{z}^\top \end{bmatrix}$.

Por fim, cada *landmark* é transladado até a origem (subtraindo $\mathbf{p}_0$) e multiplicado por R:

$$\tilde{\mathbf{p}}_i = R\left(\mathbf{p}_i - \mathbf{p}_0\right). \tag{4.2}$$

4.2.3 Ângulo de Alinhamento no Plano da Palma

Depois de fixar um referencial local na palma, ainda falta resolver a *rotação em torno do próprio eixo* Z para que todas as mãos fiquem "retas" uma em relação à outra. Escolhemos o vetor que vai do pulso até a articulação média do dedo médio,

$$\mathbf{u} = \mathbf{p}_9 - \mathbf{p}_0, \tag{4.3}$$

pois ele é aproximadamente ortogonal à linha que une os dedos indicador e mínimo, servindo como um *meridiano* natural da mão.

O ângulo de alinhamento θ é então a direção desse vetor no plano XY:

$$\theta \;=\; \mathrm{atan2}(u_y,\; u_x) \;+\; 90^\circ, \tag{4.4}$$

onde $\mathrm{atan2}(y, x)$ devolve o argumento polar do vetor (x, y) no intervalo $(-180^\circ, 180^\circ]$. O termo adicional 90° faz o dedo médio apontar para o eixo Y positivo após a rotação, definindo um *"cima"* comum para todas as capturas.

4.2.4 Aplicação da Rotação Planar

Para uniformizar o gesto, subtraímos primeiro o pulso (colocando-o na origem) e depois giramos todo o conjunto de *landmarks* em torno do eixo Z:

$$\tilde{\mathbf{p}}_i = R_Z(\theta)\,\big(\mathbf{p}_i - \mathbf{p}_0\big), \tag{4.5}$$

$$R_Z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.6}$$

Após essa etapa, quaisquer duas mãos capturadas (independentemente de como a câmera estava orientada) ficam no mesmo sistema de eixos: palma no plano $Z = 0$ e dedo médio apontando para cima.

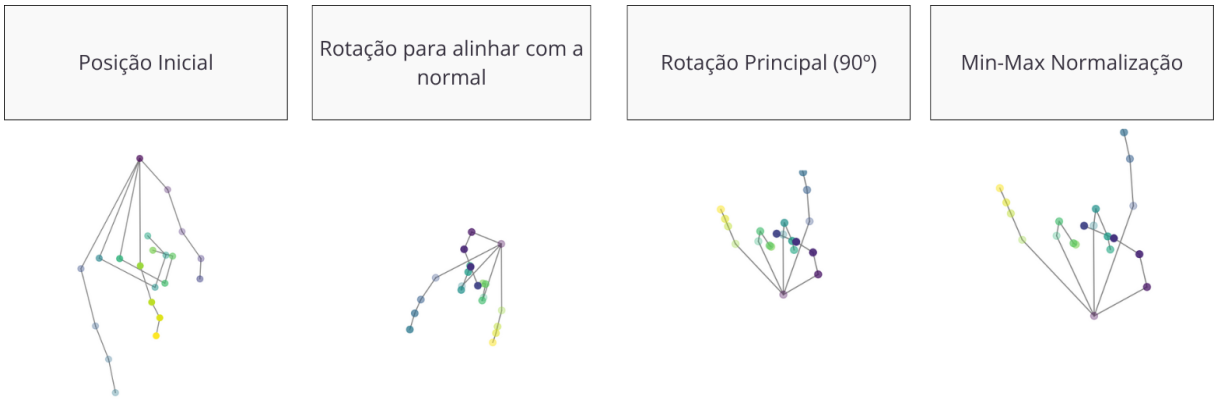
4.2.5 Escalonamento para $[0, 1]$

Com a mão já posicionada num referencial comum, resta eliminar diferenças absolutas de *tamanho*. Projetamos cada componente (x, y, z) para o intervalo $[0, 1]$ por meio de uma normalização *mínimo–máximo* feita eixo-a-eixo onde o $\min$ e o $\max$ são tomados sobre o conjunto completo de *landmarks* (21 pontos) para cada eixo separadamente. O resultado é um *bounding box* unitário cujo canto inferior esquerdo é $(0, 0, 0)$ e o canto oposto é $(1, 1, 1)$.

Esse procedimento garante que todas as configurações de mão sejam representadas consistentemente dentro de uma mesma faixa numérica, oferecendo invariância quanto às diferenças individuais no tamanho das mãos, contribuindo diretamente para a robustez e generaliza-

ção dos modelos computacionais desenvolvidos. A Figura 7 apresenta exemplos dos marcos anatômicos antes e após a aplicação do procedimento de normalização descrito.

Figura 7 – Exemplos da aplicação da normalização.



Fonte: A autora (2025)

4.3 AUMENTO ARTIFICIAL DE DADOS

Para aumentar a diversidade dos dados e aprimorar a robustez dos modelos frente a variações anatômicas e ambientais, técnicas específicas de aumento artificial (*data augmentation*) foram aplicadas ao conjunto de treinamento. Essas técnicas introduzem variações sintéticas nas representações dos gestos, simulando diferentes condições práticas que podem ocorrer em ambientes reais.

4.3.1 Rotação dos Dedos

Para ampliar a variedade dos dados de treino sem comprometer a anatomia da mão, fazemos leves rotações independentes somente nas pontas de cada dedo. Essas rotações acontecem em um único plano (eixo Z fixo), mantendo a base do dedo parada; assim, a junta principal que liga o dedo à mão continua sem se mover. A Tabela 2 traz uma lista de quais os marcos podem ser movidos em conjunto para cada dedo de forma que respeite minimamente o comportamento anatômico de uma mão.

Para cada dedo sorteia-se um limite $\theta_{\max} \in \{1^\circ, \ldots, 10^\circ\}$ e, em seguida, um ângulo $\theta \sim \mathcal{U}(-\theta_{\max}, \theta_{\max})$. Portanto o desvio mínimo é 1° e o máximo 10°.

Sejam $\mathbf{p}_b, \mathbf{p}_m, \mathbf{p}_t \in R^3$ (os pontos base, intermediário e ponta). O vetor $\mathbf{v} = \mathbf{p} - \mathbf{p}_b$ de cada ponto acima da base é girado por

Tabela 2 – Dedos e marcos considerados. Os índices seguem a convenção do MediaPipe apresentada na Figura 4.

Dedo	Base (b)	Intermediário (m)	Ponta (t)
polegar	2	3	4
indicador	6	7	8
médio	10	11	12
anelar	14	15	16
mínimo	18	19	20

Fonte: A autora (2025)

$$R_{XY}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{4.7}$$

resultando em

$$\begin{aligned} \mathbf{p}'_m &= \mathbf{p}_b + R_{XY}(\theta)\big(\mathbf{p}_m - \mathbf{p}_b\big), \\ \mathbf{p}'_t &= \mathbf{p}_b + R_{XY}(\theta)\big(\mathbf{p}_t - \mathbf{p}_b\big). \end{aligned} \tag{4.8}$$

Somente $\mathbf{p}_m$ e $\mathbf{p}_t$ são alterados; $\mathbf{p}_b$ permanece intacto, garantindo coerência com a cinemática real da mão.

4.3.2 Adição de Ruído

Com o objetivo de simular condições realísticas de captura e aumentar a robustez do modelo frente a ruídos típicos encontrados em cenários práticos, foram adicionadas perturbações aleatórias às coordenadas dos marcos anatômicos extraídos. Cada ponto anatômico recebe um deslocamento aleatório ϵ, cujos componentes são gerados independentemente por meio de uma distribuição uniforme dentro de um intervalo controlado, definido por $\delta \in [0,001; 0,005]$, conforme representado pela Equação 4.9:

$$\mathbf{p}' = \mathbf{p} + \epsilon, \quad \epsilon \sim \mathcal{U}(-\delta,\, \delta) \tag{4.9}$$

Essa técnica simula condições adversas frequentemente encontradas em aplicações reais, tais como variações na qualidade da captura (baixa resolução, movimentos indesejados) e interferências causadas por mudanças nas condições de iluminação ou outros fatores ambientais. Associada a estratégias de normalização espacial baseadas em fatores de escala e alinhamento

geométrico (TURNER; SMITH, 2023; CHUNG et al., 2023), essa abordagem contribui diretamente para aumentar a robustez, confiabilidade e capacidade de generalização dos modelos desenvolvidos, resultando em maior eficácia prática do sistema proposto.

4.4 ARQUITETURA DO MODELO

A arquitetura computacional proposta foi baseada em uma rede neural totalmente conectada (FC) e foi projetada especificamente para equilibrar eficiência, simplicidade estrutural e capacidade de generalização em contextos reais de reconhecimento automático de gestos. A escolha dessa arquitetura foi motivada pela utilização de representações compactas baseadas em marcos anatômicos (*landmarks*) das mãos, que permitem dispensar modelos computacionais mais complexos, possibilitando uma boa eficiência em tempo real em dispositivos com recursos limitados.

Cada bloco da arquitetura é composto por camadas selecionadas e otimizadas, visando garantir maior estabilidade durante o treinamento e minimizar problemas como o sobreajuste, resultando em modelos mais generalizáveis. Os componentes principais incluem:

- **Batch Normalization**: Normaliza as ativações intermediárias do modelo, estabilizando a distribuição dos dados durante o treinamento, acelerando a convergência dos algoritmos e reduzindo a variância interna dos dados (IOFFE; SZEGEDY, 2015).
- **Função de ativação ReLU (*Rectified Linear Unit*)**: Introduz não-linearidade às camadas intermediárias, favorecendo a extração de características relevantes e acelerando a convergência dos modelos (NAIR; HINTON, 2010).
- **Dropout**: Implementado com uma taxa de 0,4, essa técnica desativa aleatoriamente neurônios durante o treinamento, prevenindo efetivamente o sobreajuste e assegurando maior capacidade de generalização do modelo (SRIVASTAVA et al., 2014).

A camada final consiste em uma camada Densa seguida pela aplicação da função *Softmax*, que gera probabilidades normalizadas associadas a cada classe de gesto. Essa etapa possibilita a classificação direta e eficiente das configurações das mãos representadas pelo SignWriting, assegurando interpretações claras e precisas do sistema proposto.

4.5 AMBIENTE EXPERIMENTAL

O ambiente experimental foi elaborado com o objetivo de avaliar tanto a eficácia preditiva quanto a eficiência computacional do modelo proposto, buscando simular cenários representativos das condições práticas de uso. As subseções a seguir descrevem detalhadamente os conjuntos de dados utilizados, as métricas de avaliação adotadas e a configuração do ambiente computacional empregado.

4.5.1 Conjuntos de Dados

4.5.1.1 Conjunto de Treinamento

O conjunto de treinamento foi elaborado utilizando como base principal imagens provenientes do catálogo oficial do SignWriting[2], garantindo que as amostras estejam rigorosamente alinhadas à notação visual padronizada. Foram selecionadas três imagens diferentes (visões frontal, lateral esquerda e lateral direita) para cada um dos 261 gestos distintos considerados, resultando inicialmente em 783 imagens representativas.

Adicionalmente, para aumentar a diversidade e robustez dos dados utilizados no treinamento, foram incorporadas 25 amostras extras por classe, provenientes das partições de treinamento de 16 conjuntos de dados publicamente disponíveis, que serão apresentados na subseção seguinte. Nos casos em que os autores dos conjuntos disponibilizavam divisões explícitas em treino e teste, utilizamos exclusivamente a partição de treino. Quando tais divisões não estavam presentes, uma fração do conjunto original de teste foi separada e destinada ao treinamento, de forma estratificada, assegurando equilíbrio entre classes e consistência metodológica.

Ademais, visando aumentar significativamente a capacidade de generalização do modelo, técnicas de aumento sintético e normalização espacial foram aplicadas às amostras originais. Cada imagem foi submetida a 25 rotações artificiais e aleatórias das articulações dos dedos, simulando variações anatômicas naturais, além de 25 perturbações sintéticas de ruído, replicando condições realísticas frequentemente encontradas em ambientes reais, tais como baixa resolução das imagens ou variações nas condições de iluminação. Esses procedimentos introduziram variações controladas nas representações, preservando cuidadosamente as características fundamentais dos gestos.

[2] <https://www.signwriting.org/>

Para a validação durante o treinamento, cada conjunto de dados de treino foi dividido de forma aleatória em duas partições: 80% das amostras foram destinadas ao treinamento e 20% à validação. Essa estratégia foi preferida em relação ao uso de validação cruzada (*k-fold*), pois a maioria dos conjuntos já possui partições de teste previamente definidas pelos autores, o que inviabiliza a aplicação uniforme de *k-fold* em todos os cenários. Além disso, o custo computacional de treinar múltiplos *folds* em 16 bases distintas seria desproporcional, dado o foco do estudo em avaliar generalização entre múltiplas línguas de sinais e não em otimizar desempenho em um único corpus.

Os parâmetros finais utilizados nas técnicas de aumento artificial dos dados foram determinados com base em um estudo sistemático de ablação, cujos detalhes metodológicos e resultados quantitativos são discutidos na Seção 5.2.

4.5.1.2 Conjuntos de dados de teste.

Para a avaliação final, utilizamos 16 conjuntos de teste originais fornecidos pelos autores das bases. Como cada conjunto foi originalmente anotado em sistemas ou notações diferentes, realizamos um mapeamento manual das classes para a nomenclatura de SignWriting, garantindo consistência na comparação. Esse alinhamento exigiu analisar o catálogo oficial de SignWriting e associar cada gesto à representação equivalente, com base em semelhanças visuais e estruturais.

A Tabela 3 resume as principais características de cada conjunto de dados selecionado, que em conjunto totalizam 132 classes únicas: NUS Hand Posture dataset I (PISHARADY; VADAKKEPAT; POH, 2014), NUS Hand Posture dataset II (PISHARADY; VADAKKEPAT; LOH, 2012), OUHANDS (MATILAINEN et al., 2016), ASL Digits (MAVI, 2021), Indian Alphabet (SONAWANE, 2020), HAGRID (KAPITANOV et al., 2024), HG14 (GüLER; YüCEDAğ, 2021), LSA16 handshapes (RONCHETTI et al., 2016), Pugeault (PUGEAULT; BOWDEN, 2011), ArSL21L (GOCHOO, 2022), ASL Alphabet (NAGARAJ, 2018), KU-BdSL (JIM et al., 2023), PSL (IMRAN et al., 2021), Bengali Alphabet (RAFI et al., 2019), PHOENIX-14 Handshapes (KOLLER; NEY; BOWDEN, 2016) e LSWH100 (LOBO-NETO; PEDRINI, 2024).

Exceções: em particular, o conjunto HAGRID não dispunha de partição de teste. Trata-se de um banco extenso com aproximadamente 500 mil imagens; desse total, extraímos uma amostra balanceada de 13 mil imagens (mil por classe) para compor o conjunto de teste, além de selecionar 25 imagens adicionais por classe para compor o treinamento. Essa adaptação

Tabela 3 – Conjuntos de dados de língua de sinais utilizados nos testes, apresentando informações sobre o número de classes, tamanhos das imagens e quantidade de amostras destinadas aos testes. As línguas de sinais suportadas pelos conjuntos de dados são: ASL (*American Sign Language*), ISL (*Indian Sign Language*), LSA (*Argentine Sign Language*), ArSL (*Arabic Sign Language*), BdSL (*Bengali Sign Language*), PSL (*Pakistan Sign Language*), DGS (*German Sign Language*) e LIBRAS (Língua Brasileira de Sinais).

Nome	Linguagem de Sinal	Classes	Tamanho das imagens	Quantidade de amostras usadas para teste
NUS Hand Posture dataset I	Não definida	9	160x120	241
NUS Hand Posture dataset II	Não definida	9	160x120	2.000
OUHANDS	Não definida	10	640x480	1.000
ASL Digits	ASL	10	100x100	2.062
Indian Alphabet	ISL	13	128x128	15.600
HAGRID	Não definida	13	512x683	13.000
HG14	Não definida	14	256x256	14.000
LSA16 handshapes	LSA	15	640x480	800
Pugeault	ASL	21	87x124	12.547
ArSL21L	ArSL	21	416x416	14.202
ASL Alphabet	ASL	23	200x200	28
KU-BdSL	BdSL	25	3024x4032	1.500
PSL	PSL	31	640x480	1.480
Bengali Alphabet	BdSL	34	224x224	1.520
PHOENIX-14 Handshapes	DGS	44	93x132	1.837
LSWH100	LIBRAS	100	500x500	4.000

Fonte: A autora (2025)

assegurou a viabilidade de uso do HAGRID de forma compatível com os demais experimentos.

4.5.2 Métricas de Avaliação

Com o intuito de avaliar tanto a eficácia quanto a viabilidade operacional do modelo proposto, foram definidas quatro métricas principais. Essas métricas foram selecionadas para abranger aspectos críticos como precisão do reconhecimento, robustez em contextos multi-classes e desempenho computacional para aplicações em tempo real:

- **Acurácia**: Mede a proporção total de predições corretamente classificadas pelo modelo em relação ao total de predições realizadas, fornecendo uma avaliação direta da eficácia geral do sistema proposto.
- **F1-Score**: Combina as métricas de precisão (*precision*) e revocação (*recall*) em uma

única medida harmônica, sendo especialmente relevante para avaliar o desempenho do modelo em conjuntos com classes desbalanceadas ou com diferentes níveis de dificuldade.

- **Tempo de Inferência**: Corresponde ao tempo médio, medido em milissegundos, requerido pelo modelo para processar uma imagem e fornecer a respectiva predição. Essa métrica é especialmente importante para garantir a viabilidade prática do sistema em dispositivos com restrições computacionais, permitindo aplicações efetivas em tempo real.

- **Taxa de inferência (*Throughput*)**: Refere-se ao número médio de imagens processadas pelo modelo por unidade de tempo, tipicamente em quadros por segundo (*frames per second - FPS*). Esta métrica é crucial para avaliar o desempenho computacional e a escalabilidade do sistema, principalmente em aplicações em larga escala ou cenários que exigem respostas rápidas.

Para garantir robustez estatística dos resultados obtidos, todas as métricas foram calculadas com base em múltiplas execuções independentes (10 repetições). Os resultados finais foram reportados como médias acompanhadas por intervalos de confiança com nível de significância de 95%, permitindo assim uma análise estatística da variabilidade dos resultados obtidos.

4.5.3 Configuração do Ambiente de Treinamento

As especificações detalhadas do equipamento utilizado para a execução dos experimentos são apresentadas a seguir:

- CPU: Intel Core i7-12700H, 12ª geração (14 núcleos)

- Memória RAM: 16 GB DDR4

A configuração dos hiperparâmetros e da arquitetura do modelo foi definida utilizando técnicas de otimização automática, especificamente Otimização Bayesiana e *Random Search*. As decisões finais sobre a estrutura e hiperparâmetros foram obtidas após experimentos preliminares, detalhados na Seção 5. Os detalhes dessa configuração são apresentados abaixo:

Arquitetura da Rede Neural

- Número de camadas ocultas: 4 camadas totalmente conectadas.
- Função de ativação: ReLU (*Rectified Linear Unit*) para introdução de não-linearidade.
- Dropout: Implementado com taxa de 0,4 para prevenir sobreajuste.

4.5.3.1 Hiperparâmetros e Otimização

- **Otimizador**: Adam (*Adaptive Moment Estimation*), escolhido devido ao seu desempenho superior e ajuste dinâmico eficiente da taxa de aprendizado durante o treinamento (KINGMA; BA, 2014).
- **Função de perda**: *Categorical Cross-Entropy*, ideal para problemas de classificação multiclasse, amplamente adotada na literatura da área (BISHOP, 2006).
- **Taxa de aprendizado inicial**: 0,001, ajustada dinamicamente utilizando um *Learning Rate Scheduler* de decaimento exponencial (taxa de decaimento = 0,99 por época).
- **Tamanho do lote (*batch size*)**: 32.
- **Número máximo de épocas**: 100, com critério de parada antecipada (*Early Stopping*) baseado no desempenho do conjunto de validação para garantir eficiência computacional e evitar sobreajuste.

5 RESULTADOS

Neste capítulo são apresentados e discutidos os resultados experimentais obtidos com a metodologia proposta. As análises realizadas contemplam aspectos essenciais como eficácia preditiva, robustez a variações ambientais e eficiência computacional da abordagem em diversos cenários experimentais. Para isso, foram realizados experimentos comparativos abrangentes em múltiplos conjuntos de dados, visando demonstrar a capacidade do modelo em generalizar efetivamente para diferentes contextos linguísticos e condições práticas.

Além das análises comparativas de desempenho em diferentes cenários, são discutidos resultados detalhados obtidos através de estudos sistemáticos de ablação, cujo objetivo é avaliar quantitativamente o impacto de cada etapa do processo metodológico proposto. Esses estudos permitem compreender melhor a contribuição específica das técnicas de normalização espacial, aumento sintético dos dados e componentes arquiteturais do modelo.

Por fim, são apresentados experimentos específicos sobre a robustez do método frente a variações nas condições ambientais e diferentes níveis de restrição computacional, fornecendo evidências sobre a eficiência e aplicabilidade prática da abordagem em dispositivos com recursos limitados, cenário crucial para aplicações reais de reconhecimento automático de gestos.

5.1 COMPARAÇÃO COM MÉTODOS DO ESTADO DA ARTE

Inicialmente, foram conduzidos experimentos quantitativos comparando o desempenho do método proposto com abordagens consagradas na literatura (*estado da arte*), utilizando múltiplos conjuntos de dados amplamente reconhecidos na área de reconhecimento automático de gestos, incluindo NUS I, NUS II, OUHANDS, LSA16 e PHOENIX-14 Handshapes. Os resultados obtidos nestes experimentos são sintetizados detalhadamente na Tabela 4, permitindo avaliar diretamente o desempenho relativo da abordagem proposta frente a métodos previamente estabelecidos.

Entre os conjuntos avaliados, destaca-se o LSWH100 que é um novo conjunto de dados que contêm representações sintéticas de gestos anotados diretamente com símbolos do SignWriting. Trata-se de um conjunto pioneiro, ainda não explorado em estudos anteriores, permitindo avaliar especificamente a eficácia do método proposto em contextos baseados explicitamente nessa notação visual.

Cumpre ressaltar que, em todos os experimentos, o protocolo manteve-se idêntico: empregou-se a mesma arquitetura de rede, os mesmos hiperparâmetros e uma base inicial de treinamento comum. Para cada conjunto de dados, foram adicionadas 25 amostras do conjunto de treino do respectivo conjunto de dados, o que assegura que eventuais variações de desempenho reflitam, primordialmente, as particularidades de cada conjunto de dados.

Em síntese, mesmo não atingindo o melhor resultado em todos os conjuntos de dados, o método mantém desempenho competitivo em uma variedade de bases públicas. Essa abrangência, aliada à simplicidade arquitetural, ao baixo tempo de inferência e a pequena quantidade de amostras reais usadas para treinamento, sustenta a viabilidade do sistema em ambientes de produção que envolvem múltiplas línguas de sinais, diferentes condições de filmagem e restrições de *hardware*.

A Figura 8 apresenta as matrizes de confusão normalizadas obtidas nos experimentos realizados com os 16 conjuntos de dados distintos. Nelas, é possível observar detalhadamente a capacidade do modelo em discriminar corretamente entre diferentes classes de gestos, com predominância acentuada de valores elevados na diagonal principal indicando um alto desempenho preditivo e robustez significativa. Esses resultados confirmam a eficácia prática do método proposto na identificação correta dos gestos estáticos de mãos, corroborando claramente a sua capacidade de generalização e robustez em contextos variados.

Tabela 4 – Comparação quantitativa entre o método proposto e diversas abordagens de referência em diferentes conjuntos de dados de reconhecimento de gestos. Os métodos comparados incluem modelos amplamente utilizados, como CNN, Redes de Prototipagem (ProtoNet, do inglês *Prototypical Networks*), Redes Neurais Convolucionais com Cápsulas (CCNN, do inglês *Convolutional Capsule Neural Network*), DenseNet (do inglês *Densely Connected Convolutional Networks*), VGG16 (do inglês *Visual Geometry Group 16*), *You Only Look Once* versão 8.0 (YOLOv8) e ViT. Para cada conjunto de dados, a tabela apresenta a acurácia média obtida por cada método. No caso da nossa abordagem, os resultados são exibidos com um intervalo de erro calculado a partir de *bootstrap* de 10 execuções independentes. Os traços (–) indicam que o respectivo trabalho da literatura não reportou resultados naquele conjunto específico, já que a maioria dos estudos foi avaliada apenas em uma, duas ou três bases, e não em todas as utilizadas neste estudo. Dessa forma, a comparação deve ser entendida como parcial: cada método da literatura é contrastado com o proposto apenas nos conjuntos em comum.

Método	NUS I	NUS II	OUHANDS	ASL Digits
CNN (GUPTA et al., 2022)	0,9943	-	-	-
2RCNN (SAHOO et al., 2022)	-	0,9480	-	-
CNN (KUMAR; SURESH; DINESH, 2022)	-	-	0,8757	-
CNN (GUPTA et al., 2022)	-	-	-	**0,9906**
Este estudo	**1,0000 ± 0,0000**	**0,9871 ± 0,0056**	**0,9818 ± 0,0123**	0,9902 ± 0,0053

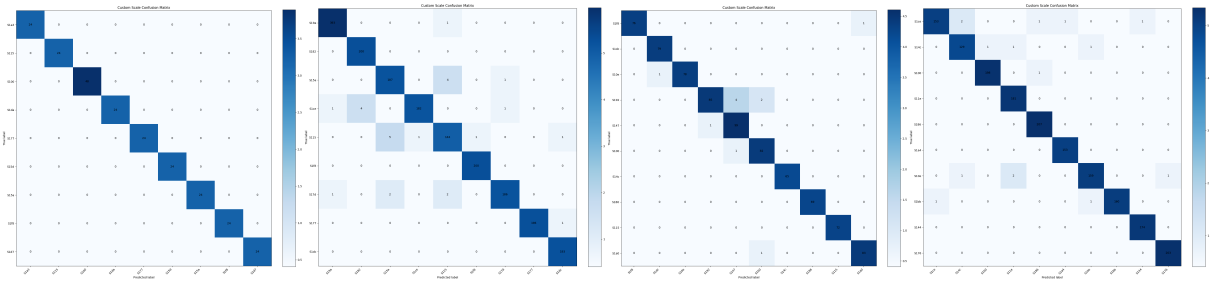
Método	Indian Alphabet	HAGRID	HG14	LSA16
FC (FALLAH et al., 2024)	**0,9995**	-	-	-
CNN (MENON; SRUTHI; LIJIYA, 2022)	0,9986	-	-	-
Densenet201 (PADHI; DAS, 2022)	-	0,9755	-	-
CNN (MISHRA et al., 2023)	-	**0,9921**	-	-
CCNN (GüLER; YüCEDAğ, 2021)	-	-	0,8739	-
ResNet50 (AWALUDDIN; CHAO; CHIOU, 2023)	-	-	**0,9747**	-
ProtoNet (RONCHETTI et al., 2023)	-	-	-	**0,9838**
VGG16 (QUIROGA et al., 2017)	-	-	-	0,9592
Este estudo	0,9968 ± 0,0009	0,9549 ± 0,0022	0,9685 ± 0,0045	0,8439 ± 0,0584

Método	Pugeault	ArSL21L	ASL Alphabet	KU-BdSL
ViT (ZHANG et al., 2023)	**0,9653**	-	0,9944	-
YOLOv8 (ALAMRI et al., 2024)	-	**0,9799**	-	-
FC (FALLAH et al., 2024)	-	-	0,9940	-
ViT (ALSHARIF et al., 2023)	-	-	**0,9998**	-
VGG16 (SURJO et al., 2023)	-	-	-	0,9800
Este estudo	0,9483 ± 0,0068	0,9598 ± 0,0046	0,9167 ± 0,0986	**0,9865 ± 0,0091**

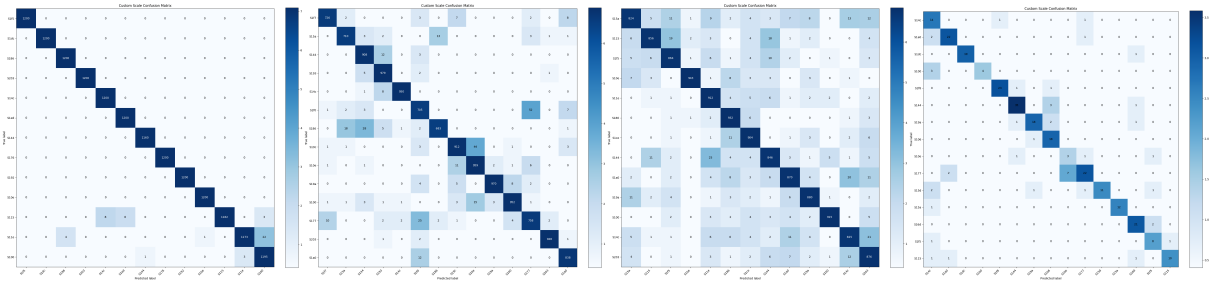
Método	PSL	Bengali Alphabet	PHOENIX-14 Handshapes	LSWH100
CNN (AROOJ et al., 2024)	**0,9874**	-	-	-
FC (FALLAH et al., 2024)	-	**0,9996**	-	-
ANN (IRVANIZAM; HORATIUS; SOFYAN, 2023)	-	0,9841	-	-
DenseNet (RONCHETTI et al., 2023)	-	-	**0,9605**	-
Este estudo	0,9819 ± 0,0062	0,7258 ± 0,0263	0,8553 ± 0,0175	0,9098 ± 0,0085

Fonte: A autora (2025)

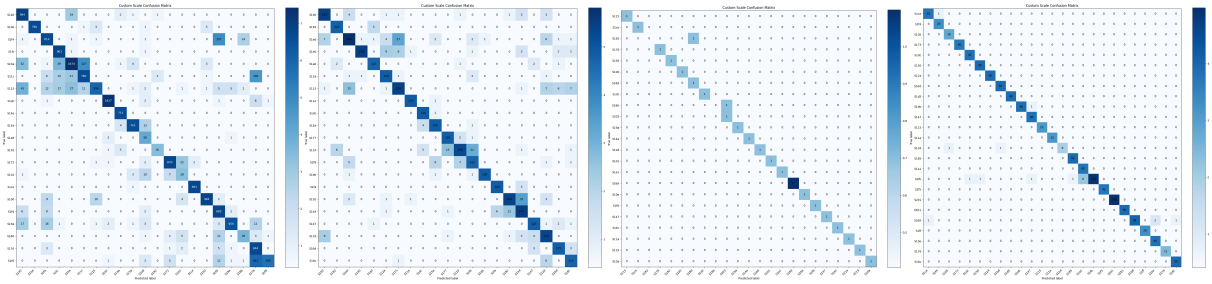
Figura 8 – Matrizes de confusão normalizadas para os 16 experimentos realizados em diferentes conjuntos de dados de reconhecimento de gestos. Cada matriz ilustra o desempenho do modelo em termos de acurácia de classificação entre classes, com maior intensidade de cor ao longo da diagonal indicando melhor precisão de predição.



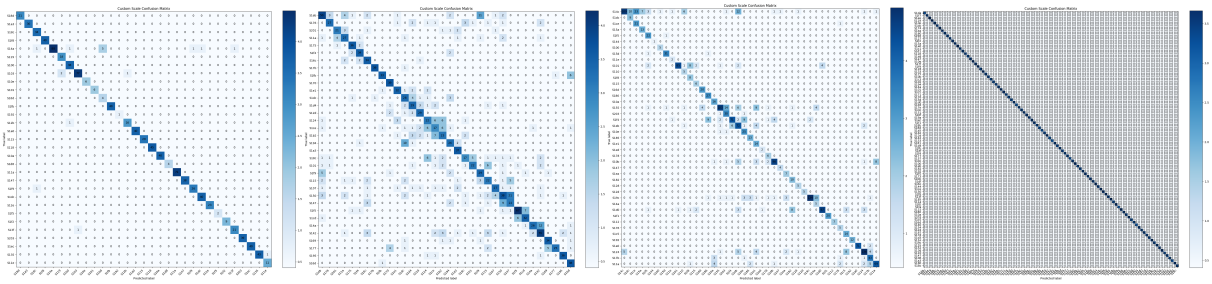
(a) NUS I (b) NUS II (c) OUHANDS (d) ASL Digits



(e) Indian Alphabet (f) HG14 (g) HAGRID (h) LSA16



(i) Pugeault (j) ArSL21L (k) ASL Alphabet (l) KU-BdSL



(m) PSL (n) Bengali Alphabet (o) PHOENIX-14 (p) LSWH100

Fonte: A autora (2025)

5.2 ESTUDO DE ABLAÇÃO

Com o objetivo de avaliar detalhadamente a contribuição individual de cada componente metodológico no desempenho geral do modelo proposto, foi realizado um estudo sistemático de ablação. Essa abordagem permite identificar claramente quais etapas metodológicas são essenciais para o funcionamento eficaz do sistema, bem como entender precisamente o impacto isolado e combinado de estratégias como normalização espacial, aumento de dados sintéticos e diferentes modelos de aprendizado utilizados (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Para os experimentos de ablação adotou-se exclusivamente o HAGRID, um corpus de gestos genéricos de mão que não está vinculado a nenhuma língua de sinais específica. Esta escolha deve-se a três fatores principais: oferece um conjunto de teste volumoso com 13.000 amostras; exibe qualidade de imagem superior a coleções mais antigas e já é largamente empregado na literatura, o que facilita comparações diretas.

5.2.1 Impacto das Arquiteturas e Modelos de Aprendizado

O primeiro experimento realizado visou avaliar quantitativamente o impacto direto das diferentes arquiteturas e algoritmos de aprendizado sobre a tarefa específica de classificação automática de gestos. Todos os classificadores avaliados foram inseridos no mesmo fluxo ilustrado na Figura 5. Os valores resultantes da normalização são então concatenados em um único vetor de características, que serve de entrada aos modelos comparados.

Foram comparadas múltiplas abordagens amplamente adotadas na literatura, abrangendo desde métodos clássicos de aprendizado de máquina até arquiteturas profundas mais complexas, conforme detalhado na Tabela 5. Entre as técnicas avaliadas encontram-se FC, Redes Neurais Convolucionais Unidimensionais (CONV1D, do inglês *1D Convolutional Neural Networks*), Florestas Aleatórias (RF, do inglês *Random Forest*), Máquinas de Vetores de Suporte (SVM, do inglês, *Support Vector Machines*), Algoritmo dos K-Vizinhos Mais Próximos (KNN, do inglês *K-Nearest Neighbors*), Regressão Logística (LR, do inglês *Logistic Regression*), *Gradient Boosting* (GBC) e AdaBoost (*Adaptive Boosting*).

Adicionalmente, foi explorada uma arquitetura baseada em Redes Totalmente Conectadas com blocos residuais, combinada a um *embedding* pré-treinado disponibilizado pelo Google (denominado *FC + embedder*). Essa abordagem específica busca combinar a eficiência computacional das redes totalmente conectadas com a capacidade aprimorada de representação

dos *embeddings* pré-treinados, visando capturar relações mais complexas e robustas presentes nos dados.

Para garantir uma análise robusta e abrangente, em cada caso também foram avaliadas diferentes combinações de estratégias de normalização espacial e técnicas de aumento de dados sintéticos, como *Rotação dos Dedos* e *Adição de Ruído*, permitindo uma avaliação rigorosa do impacto isolado e combinado dessas estratégias sobre o desempenho preditivo do modelo.

Tabela 5 – Comparação do desempenho de diferentes modelos em termos de acurácia e *F1-score*, considerando várias combinações de normalização e métodos de aumento de dados no conjunto de dados HAGRID.

Sem métodos de normalização e sem aumento de dados		
Modelo	Acurácia	F1
FC	0,1106 ± 0,0059	0,0421 ± 0,0032
FC + embedder	**0,6564 ± 0,0119**	**0,6212 ± 0,0131**
CONV1D	0,0899 ± 0,0069	0,0234 ± 0,0029
RF	0,1214 ± 0,0034	0,0756 ± 0,0033
SVM	0,0841 ± 0,0069	0,0389 ± 0,0046
GBC	0,0766 ± 0,0058	0,0157 ± 0,0021
Adaboost	0,0994 ± 0,0055	0,0607 ± 0,0057
KNN	0,0946 ± 0,0049	0,0616 ± 0,0045
LR	0,1118 ± 0,0033	0,0506 ± 0,0019
Com normalização e sem métodos de aumento		
FC	0,5147 ± 0,0082	0,476 ± 0,0104
FC + embedder	0,5317 ± 0,0105	0,5081 ± 0,0107
CONV1D	0,1591 ± 0,0086	0,0566 ± 0,0052
RF	0,6023 ± 0,0158	0,5736 ± 0,0169
SVM	**0,6312 ± 0,0084**	**0,5909 ± 0,0088**
GBC	0,3364 ± 0,0103	0,3253 ± 0,0098
Adaboost	0,1138 ± 0,0063	0,098 ± 0,0066
KNN	0,2667 ± 0,0068	0,2362 ± 0,0106
LR	0,5861 ± 0,0056	0,5388 ± 0,0067
Com normalização e com aumento de "Rotação dos Dedos"		
FC	**0,7305 ± 0,0098**	**0,7254 ± 0,0106**
FC + embedder	0,7042 ± 0,0107	0,7054 ± 0,011
CONV1D	0,5279 ± 0,0078	0,5126 ± 0,0075
RF	0,6433 ± 0,0131	0,6296 ± 0,0137
SVM	0,6534 ± 0,0076	0,6534 ± 0,0081
GBC	0,3389 ± 0,0116	0,3303 ± 0,0111
Adaboost	0,1206 ± 0,0089	0,0684 ± 0,0069
KNN	0,5968 ± 0,0123	0,5962 ± 0,0124
LR	0,6456 ± 0,0088	0,6356 ± 0,0107
Com normalização e com aumentos de "Rotação dos Dedos" e "Ruído"		
FC	**0,757 ± 0,0085**	**0,7531 ± 0,0083**
FC + embedder	0,6895 ± 0,0117	0,6883 ± 0,0127
CONV1D	0,4893 ± 0,0095	0,4718 ± 0,0094
RF	0,6263 ± 0,0122	0,6095 ± 0,013
SVM	0,6579 ± 0,0082	0,6524 ± 0,0072
GBC	0,3142 ± 0,0126	0,3126 ± 0,0128
Adaboost	0,2131 ± 0,0125	0,1499 ± 0,0084
KNN	0,6002 ± 0,0091	0,5995 ± 0,0085
LR	0,6489 ± 0,0092	0,6391 ± 0,0082

Fonte: A autora (2025)

5.2.2 Impacto de Diferentes Conjuntos de Treinamento

Um segundo experimento de ablação foi realizado com o objetivo específico de avaliar o impacto direto que diferentes composições do conjunto de treinamento têm sobre o desempenho final do modelo proposto. Para garantir consistência metodológica e comparabilidade direta dos resultados, foi utilizada a configuração ótima identificada no estudo anterior, incluindo técnicas de normalização espacial ativa e estratégias de aumento sintético de dados ("Rotação dos Dedos" e "Adição de Ruído").

Os resultados obtidos com esse experimento são apresentados detalhadamente na Tabela 6, considerando três configurações distintas de composição do conjunto de treinamento, descritas a seguir:

- **SignWriting (oficial)**: Modelo treinado exclusivamente com o conjunto oficial de imagens de SignWriting. Esta configuração serve como referência direta (*baseline*), permitindo avaliar especificamente o desempenho do método na representação padronizada original, sem a influência de outras variações externas.

- **SignWriting + LSWH100**: Modelo treinado utilizando a combinação do conjunto oficial SignWriting com o conjunto LSWH100, contendo 100 classes de gestos representadas diretamente em SignWriting. Essa configuração foi projetada para avaliar explicitamente o impacto da inclusão do LSWH100 sobre a robustez, generalização e desempenho geral do método em condições variadas.

- **SignWriting + Amostra do Experimento**: Modelo treinado com a combinação do conjunto oficial do SignWriting e pequenas amostras específicas extraídas do conjunto do experimento sendo avaliado. Foram utilizadas 25 amostras adicionais por classe, provenientes do respectivo conjunto quando disponíveis. Para os conjuntos de dados sem partições pré-definidas para treinamento, separou-se uma amostra de 25 exemplos por classe para o treinamento, preservando as demais para teste. Essa configuração foi adotada para avaliar o impacto prático direto de complementar o treinamento com amostras específicas do domínio-alvo, visando aumentar a robustez e o desempenho preditivo do modelo em condições mais próximas da aplicação prática.

Tabela 6 – Resultados de desempenho do modelo no conjunto de dados HAGRID em três configurações de treinamento, medidos por acurácia e *F1-score*, todos acompanhados de intervalos de confiança calculados por meio de *bootstrap*.

Conjunto de treinamento	Acurácia	F1
SW	0,7247 ± 0,0065	0,7166 ± 0,0058
SW + LSWH100	0,8705 ± 0,0060	0,8694 ± 0,0060
SW + Amostra do Experimento	**0,9549 ± 0,0022**	**0,9549 ± 0,0022**

Fonte: A autora (2025)

5.2.3 Impacto do Fator de Aumento e do Tamanho da Amostra

Um terceiro estudo de ablação foi realizado com o objetivo de avaliar como o tamanho inicial do conjunto de treinamento e o fator de aumento artificial dos dados (*data augmentation*) influenciam o desempenho preditivo do modelo. Mais especificamente, buscou-se identificar quantitativamente o impacto combinado dessas duas variáveis críticas na capacidade geral de generalização do método proposto.

Os resultados quantitativos obtidos nesse estudo são apresentados na Tabela 7, destacando o desempenho alcançado em termos de acurácia e *F1-score*, considerando diferentes combinações entre o fator de aumento aplicado às amostras e o número inicial de exemplos de treinamento por classe.

O fator de aumento (*augmentation factor*) refere-se diretamente ao número de amostras adicionais geradas artificialmente para cada exemplo original, utilizando técnicas previamente descritas, como Rotação dos Dedos e Adição de Ruído. Os valores considerados variaram entre 5 e 25, permitindo expor sistematicamente o modelo a diferentes graus de variabilidade controlada, com o intuito de aprimorar explicitamente a capacidade de generalização.

Além disso, foi analisado o impacto do número inicial de amostras disponíveis para treinamento (antes da aplicação das técnicas de aumento), variando também entre 5 e 25 amostras por classe. Essa avaliação permitiu compreender a interação entre a quantidade inicial de dados disponíveis e o fator de aumento artificial na capacidade preditiva e robustez do modelo.

5.2.4 Avaliação de Desempenho sob Diferentes Restrições Computacionais

Visando avaliar como diferentes restrições computacionais impactam o desempenho do modelo durante a etapa de inferência, foram realizados experimentos sistemáticos em ambientes controlados utilizando Docker. O modelo treinado foi convertido para o formato TensorFlow

Tabela 7 – Desempenho do modelo em termos de acurácia e *F1-score* no conjunto de dados HAGRID para diferentes combinações de fatores de aumento e número de amostras de treinamento. O fator de aumento, variando de 5 a 25, representa a multiplicação de amostras para elevar a variabilidade dos dados, enquanto o número de amostras de treinamento (de 5 a 25) indica a quantidade de exemplos originais utilizados antes do aumento.

Fator	Número de amostras	Acurácia	F1
5	5	0,9227 ± 0,0079	0,9231 ± 0,0077
5	10	0,9419 ± 0,0034	0,9418 ± 0,0035
5	15	0,9462 ± 0,0045	0,9465 ± 0,0044
5	20	0,9515 ± 0,0040	0,9517 ± 0,0040
5	25	0,9507 ± 0,0052	0,9509 ± 0,0051
10	5	0,9133 ± 0,0027	0,9130 ± 0,0027
10	10	0,9356 ± 0,0058	0,9354 ± 0,0058
10	15	0,9491 ± 0,0048	0,9492 ± 0,0048
10	20	0,9485 ± 0,0049	0,9486 ± 0,0050
10	25	0,9538 ± 0,0050	0,9539 ± 0,0049
15	5	0,9198 ± 0,0062	0,9200 ± 0,0062
15	10	0,9390 ± 0,0058	0,9388 ± 0,0058
15	15	0,9439 ± 0,0035	0,9441 ± 0,0035
15	20	0,9457 ± 0,0056	0,9458 ± 0,0057
15	25	0,9541 ± 0,0044	0,9543 ± 0,0044
20	5	0,9209 ± 0,0053	0,9206 ± 0,0054
20	10	0,9415 ± 0,0044	0,9416 ± 0,0043
20	15	0,9486 ± 0,0055	0,9487 ± 0,0055
20	20	0,9521 ± 0,0045	0,9522 ± 0,0045
20	25	0,9482 ± 0,0037	0,9482 ± 0,0037
25	5	0,9358 ± 0,0037	0,9359 ± 0,0037
25	10	0,9335 ± 0,0050	0,9337 ± 0,0049
25	15	0,9501 ± 0,0033	0,9502 ± 0,0033
25	20	0,9518 ± 0,0032	0,9520 ± 0,0032
25	25	**0,9554 ± 0,0056**	**0,9555 ± 0,0056**

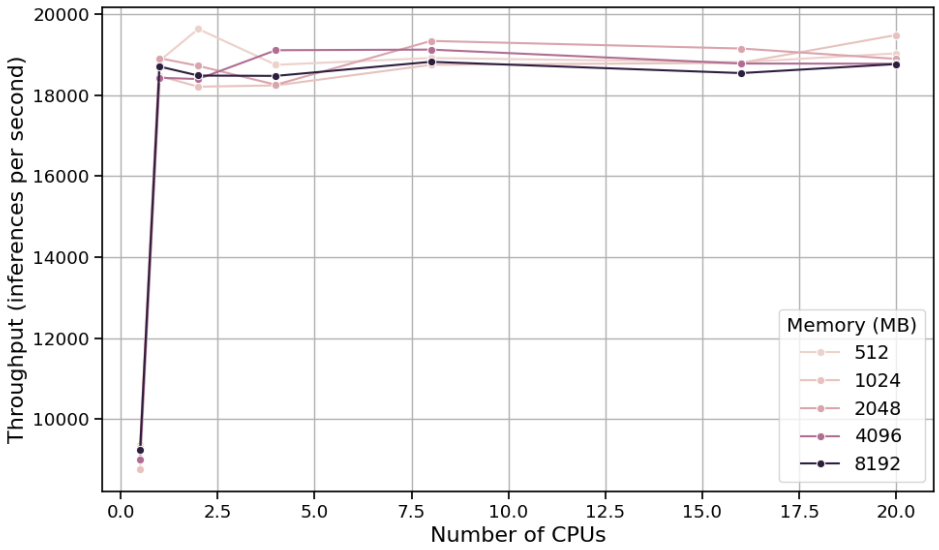
Fonte: A autora (2025)

Lite (TFLite), sem aplicação de otimizações adicionais (como quantização), resultando em um arquivo compacto de aproximadamente 3 megabytes. Essa conversão visou especificamente avaliar a capacidade prática do modelo em ambientes com recursos computacionais limitados.

Durante os testes, foram medidas métricas quantitativas como o tempo médio de inferência, a taxa de inferências por segundo (*throughput*), e o consumo médio de CPU e memória. Cada configuração testada foi repetida 30 vezes, utilizando lotes padronizados de 1.000 inferências cada, garantindo robustez estatística e replicabilidade dos resultados.

A Figura 9 detalha a relação entre o número de núcleos de CPU e o *throughput* obtido em diferentes configurações de memória, evidenciando a escalabilidade e a estabilidade operacional da abordagem. Observa-se claramente que o *throughput* aumenta proporcionalmente ao número de núcleos de CPU até atingir um ponto de saturação, no qual incrementos adicionais resultam em ganhos mínimos. Para referência, a configuração mais eficiente obteve aproximadamente 18.500 inferências por segundo, correspondendo a um tempo médio de apenas 0,00005 segundos por inferência.

Figura 9 – Relação entre número de CPUs e *throughput*

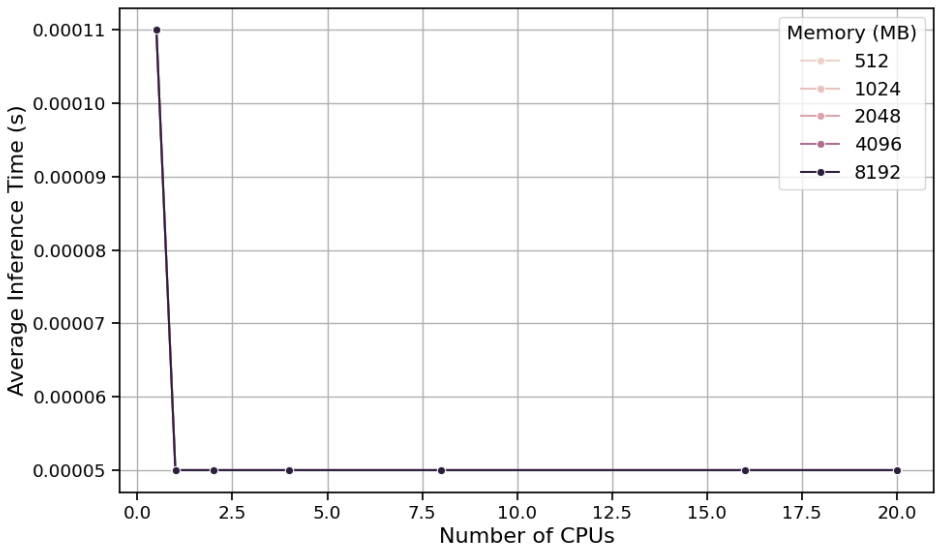


Fonte: A autora (2025)

A Figura 10 ilustra o impacto direto do número de CPUs sobre o tempo médio de inferência por amostra. Observa-se claramente que o tempo médio diminui à medida que são adicionados núcleos de CPU, até alcançar um platô em que ganhos adicionais são reduzidos. Esses resultados fornecem evidências claras sobre o equilíbrio ideal entre desempenho obtido e recursos computacionais utilizados, facilitando decisões práticas sobre alocação eficiente em aplicações reais.

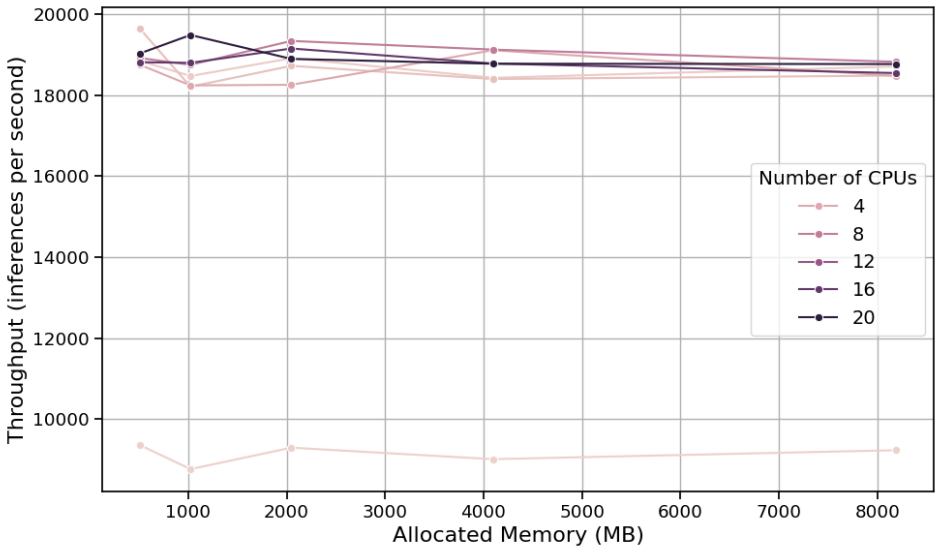
Finalmente, a Figura 11 apresenta a relação entre quantidade de memória alocada e o *throughput*, revelando desempenho constante e robusto em todas as configurações avaliadas. Esses achados demonstram a capacidade do modelo de operar eficientemente em ambientes computacionais com recursos limitados, sugerindo sua viabilidade prática mesmo em dispositivos com restrições significativas de memória.

Figura 10 – Relação entre número de CPUs e tempo médio de inferência



Fonte: A autora (2025)

Figura 11 – Relação entre memória e *throughput*



Fonte: A autora (2025)

5.2.5 Impacto de Erros de Detecção de Marcos (*Landmarks*)

Com o objetivo de compreender o impacto das falhas ou imprecisões na etapa de detecção automática dos marcos anatômicos (*landmarks*) realizada pelo Mediapipe, foi conduzida uma análise manual detalhada. Essa análise buscou quantificar o quanto os erros de detecção afetam diretamente o desempenho global do modelo proposto.

Os resultados quantitativos detalhados obtidos nessa análise estão apresentados na Ta-

bela 8, especificando a acurácia global original, o número total de erros observados, a quantidade de erros diretamente atribuíveis à detecção dos marcos pelo Mediapipe e a acurácia ajustada calculada formalmente para cada conjunto de dados.

Essa acurácia ajustada é calculada pela Equação 5.1, que procura isolar os erros diretamente relacionados à detecção de marcos pelo Mediapipe, oferecendo uma estimativa do desempenho que poderia ser alcançado caso esses erros fossem mitigados:

$$\text{Acurácia Ajustada} = \frac{N_{\text{total}} - N_{\text{mediapipe\_error}} - N_{\text{model\_error}}}{N_{\text{total}} - N_{\text{mediapipe\_error}}} \tag{5.1}$$

onde:

- N_{total}: Número total de amostras avaliadas.
- $N_{\text{mediapipe\_error}}$: Quantidade de erros atribuíveis diretamente ao Mediapipe.
- $N_{\text{model\_error}}$: Número de erros cometidos exclusivamente pelo modelo, excluindo-se as falhas provenientes diretamente do Mediapipe.

Tabela 8 – Resumo do desempenho do modelo com acurácia ajustada conforme a Equação 5.1 para cada conjunto de dados. A tabela inclui a acurácia original, a contagem total de erros, os erros atribuídos ao Mediapipe e a acurácia ajustada recalculada, que leva em consideração as falhas de detecção relacionadas ao Mediapipe.

Conjunto de dados	Acurácia	Erro total	Erros do Mediapipe	Erro Mediapipe (%)	Acurácia Ajustada
NUS II	0,9871	26	12	46,15%	0,9926
OUHANDS	0,9818	11	4	36,36%	0,9911
ASL Digits	0,9902	15	14	93,33%	0,9994
Indian Alphabet	0,9968	48	4	8,33%	0,9971
HAGRID	0,9549	544	437	80,33%	0,9907
HG14	0,9685	384	245	63,80%	0,9885
LSA16	0,8439	45	23	51,11%	0,9160
Pugeault	0,9483	346	161	46,53%	0,9720
ArSL21L	0,9598	155	48	30,97%	0,9719
ASL Alphabet	0,9167	2	2	100,0%	1,0000
KU-BdSL	0,9865	20	12	60,0%	0,9932
PSL	0,9819	17	15	88,24%	0,9979
Bengali Alphabet	0,7258	393	201	51,15%	0,8482
PHOENIX-14	0,8553	856	804	93,93%	0,9461
LSWH100	0,9098	278	86	30,94%	0,9377

Fonte: A autora (2025)

É importante enfatizar que essa métrica não substitui as medidas tradicionais de acurácia; seu objetivo específico é fornecer uma estimativa da penalização sofrida pelo modelo devido aos erros externos à classificação, provenientes exclusivamente da etapa de detecção automática.

Os resultados apresentados na Tabela 8 indicam que o número de erros do MediaPipe é elevado em alguns conjuntos, comprometendo de forma significativa a acurácia global. Isso sugere que parte considerável das discrepâncias observadas não decorre do classificador proposto, mas sim de limitações na etapa de pré-processamento. Assim, ao comparar os resultados com outros métodos da literatura, deve-se considerar que o desempenho do modelo aqui avaliado sofre restrições adicionais, pois depende integralmente da qualidade dos marcos detectados pelo MediaPipe. Em outras palavras, a acurácia reportada pode estar subestimando a real capacidade discriminativa da arquitetura, já que erros do detector são propagados ao classificador sem possibilidade de correção posterior.

Para mitigar esse problema em trabalhos futuros, algumas estratégias práticas podem ser exploradas, tais como:

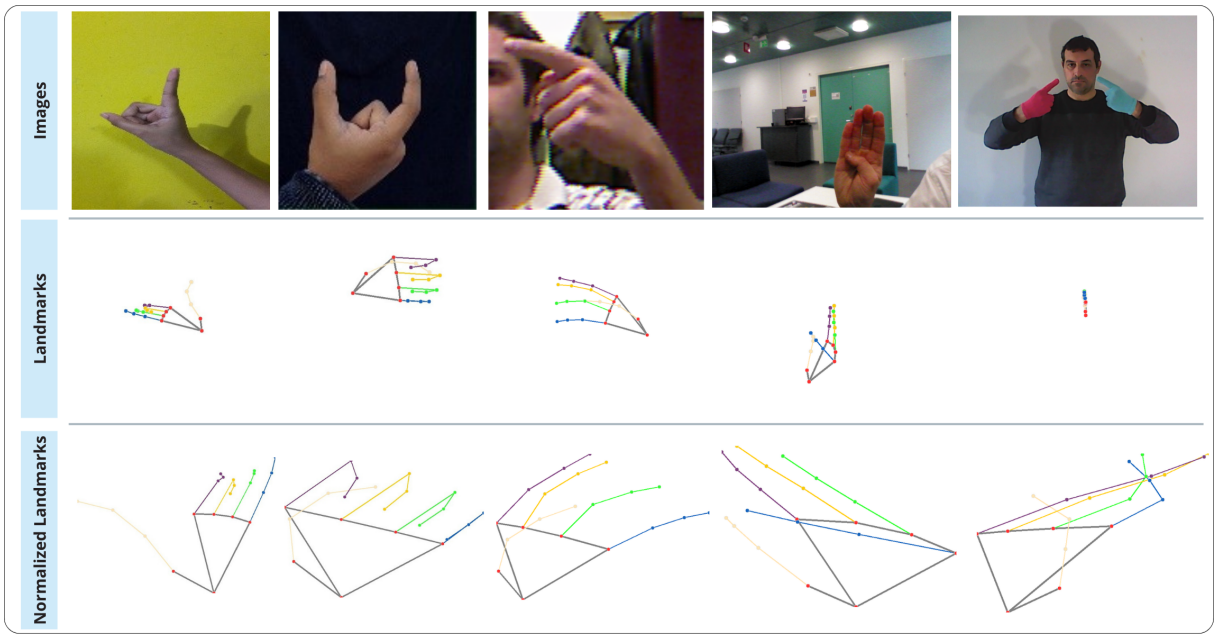
- Aplicação de filtros estatísticos de *outliers* para descartar quadros com marcos evidentemente incoerentes;
- Uso de técnicas de suavização temporal em sequências de vídeo (ex.: filtros de Kalman ou médias móveis), reduzindo a variabilidade abrupta entre quadros consecutivos;
- Calibração ou ajuste fino dos parâmetros internos do MediaPipe, de forma a adequar o detector às condições específicas de captura de cada base;
- Exploração de detectores alternativos ou modelos híbridos (ex.: OpenPose combinado ao MediaPipe), de modo a aumentar a robustez na detecção dos marcos.

Embora o MediaPipe apresente elevada eficiência e seja amplamente adotado pela comunidade de visão computacional, sua utilização em bases heterogêneas expõe limitações que, neste trabalho, se mostraram relevantes. Reconhecer o peso desses erros é fundamental para interpretar corretamente os resultados obtidos e para guiar melhorias metodológicas que aproximem o desempenho observado do desempenho potencial estimado pela acurácia ajustada.

5.3 ANÁLISE QUALITATIVA DE ERROS DE DETECÇÃO DO MEDIAPIPE

Para compreender de forma qualitativa o impacto específico das falhas na etapa de detecção automática de marcos anatômicos (*landmarks*) pelo Mediapipe sobre o desempenho geral do modelo de classificação proposto, realizou-se uma análise detalhada por meio de verificação manual. Essa análise qualitativa buscou identificar quais erros podem ser atribuídos ao Mediapipe, diferenciando-os dos erros diretamente relacionados ao próprio modelo de reconhecimento de gestos.

Figura 12 – Exemplos qualitativos de erros na detecção de marcos pelo Mediapipe em reconhecimento de gestos. A figura exibe cinco gestos distintos, cada um em três estágios: (1) a imagem original, (2) os marcos (*landmarks* detectados pelo Mediapipe e (3) os marcos normalizados utilizados como entrada para o modelo. Em cada caso, o Mediapipe falha em capturar corretamente os marcos da mão, resultando em representações desalinhadas, incompletas ou distorcidas. Esses erros de detecção, como pontas dos dedos ausentes ou posições incorretas dos dedos, introduzem ruído no processo, afetando negativamente a precisão de classificação do modelo.



Fonte: A autora (2025)

Na Figura 12, são apresentados exemplos qualitativos representativos de erros específicos cometidos pela ferramenta Mediapipe durante a detecção automática dos marcos anatômicos das mãos. Cada exemplo detalha claramente três estágios distintos do processo: a imagem original contendo o gesto real realizado, os marcos anatômicos detectados automaticamente pelo Mediapipe e, finalmente, a representação normalizada desses marcos, que corresponde à entrada fornecida ao modelo durante a etapa de classificação.

Os erros ilustrados ocorrem tipicamente quando o MediaPipe não consegue identificar cor-

retamente a posição ou orientação dos dedos, resultando em representações incorretas, distorcidas ou incompletas em relação aos gestos originais. Consequentemente, tais erros impactam diretamente e significativamente a eficácia do processo de classificação, já que o modelo passa a receber entradas ruidosas, imprecisas ou inconsistentes com o gesto originalmente realizado. Essa situação evidencia a importância crítica da precisão na etapa inicial de detecção dos marcos para a robustez global do sistema.

Essa análise qualitativa ressalta explicitamente a necessidade de técnicas mais rigorosas de pré-processamento e evidencia a importância de investigar aprimoramentos ou alternativas ao Mediapipe na etapa de detecção de marcos anatômicos.

6 CONCLUSÃO

Este trabalho apresentou o desenvolvimento e a validação experimental de um sistema automático de reconhecimento de gestos estáticos das mãos baseado no SignWriting, com o intuito específico de superar barreiras linguísticas e culturais presentes em diferentes línguas de sinais. A metodologia proposta utilizou marcos anatômicos (*landmarks*) extraídos automaticamente com o MediaPipe, técnicas robustas de normalização espacial, estratégias de aumento sintético dos dados e uma arquitetura de rede neural totalmente conectada para realizar a classificação dos gestos.

A avaliação realizada em 16 conjuntos de dados distintos, contemplando um total de 132 classes únicas de gestos, revelou resultados consistentes quanto à capacidade de generalização do método proposto, destacando o potencial do SignWriting como solução eficaz e unificadora para o reconhecimento interlinguístico de configurações de mão (*handshapes*). Além disso, o estudo evidenciou a praticidade do sistema em cenários reais e reforçou a relevância acadêmica e tecnológica do SignWriting como notação visual independente de idioma.

Contudo, foram identificadas limitações importantes que oferecem oportunidades futuras claras para aprimoramentos. A ferramenta MediaPipe, embora eficiente operacionalmente em tempo real, demonstrou vulnerabilidade em condições adversas, como baixa iluminação e gestos anatômicos complexos, influenciando diretamente a precisão final do sistema. Além disso, a abordagem focada exclusivamente em gestos estáticos não contemplou outros aspectos essenciais das línguas de sinais, especialmente movimentos contínuos e expressões faciais.

Essas limitações identificadas fornecem caminhos concretos para pesquisas futuras:

- **Aprimoramento da detecção de marcos**: Investigação de técnicas de pós-processamento, como filtragem robusta de *outliers*, suavização temporal das detecções e alternativas ou melhorias ao MediaPipe para aprimorar a robustez da detecção de marcos.
- **Reconhecimento de sinais dinâmicos**: Incorporar recursos adicionais do SignWriting relacionados a movimentos e sequências gestuais, ampliando a abrangência linguística e comunicativa do sistema.
- **Integração de expressões faciais e contexto linguístico**: Explorar métodos para integrar expressões faciais e contextuais na classificação, aumentando a completude linguística e aplicabilidade prática da abordagem.

Em síntese, os resultados obtidos fornecem evidências de que o SignWriting, em combinação com técnicas de visão computacional e aprendizado profundo, oferece um alicerce para sistemas futuros mais completos e inclusivos de tradução de línguas de sinais.

REFERÊNCIAS

ABDULLAH, A.; ALI, N.; ALI, R. H.; ABIDEEN, Z. U.; IJAZ, A. Z.; BAIS, A. American Sign Language Character Recognition using Convolutional Neural Networks. *2023 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, p. 165–169, 2023.

ABDULLAH, B. A. A.; AMOUDI, G. A.; ALGHAMDI, H. S. Advancements in Sign Language Recognition: A Comprehensive Review and Future Prospects. *IEEE Access*, v. 12, p. 128871–128895, 2024.

AL-QURISHI, M.; KHALID, T.; SOUISSI, R. Deep Learning for Sign Language Recognition: Current Techniques, Benchmarks, and Open Issues. *IEEE Access*, v. 9, p. 126917–126951, 2021.

ALABBAD, D. A.; ALSALEH, N. O.; ALAQEEL, N. A.; ALSHEHRI, Y. A.; ALZAHRANI, N. A.; ALHOBAISHI, M. K. A Robot-based Arabic Sign Language Translating System. *2022 7th International Conference on Data Science and Machine Learning Applications (CDMA)*, p. 151–156, 2022.

ALAMRI, F. S.; REHMAN, A.; ABDULLAHI, S. B.; SABA, T. Intelligent Real-Life Key-Pixel Image Detection System for Early Arabic Sign Language Learners. *PeerJ Computer Science*, v. 10, p. e2063–e2063, 2024.

ALAYED, A. Machine Learning and Deep Learning Approaches for Arabic Sign Language Recognition: A Decade Systematic Literature Review. *Sensors*, v. 24, p. 7798, 2024.

ALNABIH, A. F.; MAGHARI, A. Y. Arabic Sign Language Letters Recognition Using Vision Transformer. *Multimedia Tools and Applications*, 2024.

ALSHARIF, B.; ALALWANY, E.; IBRAHIM, A.; MAHGOUB, I.; ILYAS, M. Real-Time American Sign Language Interpretation Using Deep Learning and Keypoint Tracking. *Sensors*, v. 25, p. 2138, 2025.

ALSHARIF, B.; ALANAZI, M.; ALTAHER, A. S.; ALTAHER, A.; ILYAS, M. Deep Learning Technology to Recognize American Sign Language Alphabet Using Mulit-Focus Image Fusion Technique. *2023 IEEE 20th International Conference on Smart Communities: Improving Quality of Life using AI, Robotics and IoT (HONET)*, p. 1–6, 2023.

AROOJ, S.; ALTAF, S.; AHMAD, S.; MAHMOUD, H.; , M. . Enhancing Sign Language Recognition using CNN and SIFT: A Case Study on Pakistan Sign Language. *Journal of King Saud University - Computer and Information Sciences*, v. 36, p. 101934–101934, 2024.

AWAD, A. D.; KOYUNCU, H. Hand Gesture Recognition for Interactive Media Player Using CNN and Image Classification. *2022 International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, v. 30, p. 753–756, 2022.

AWALUDDIN, B.-A.; CHAO, C.-T.; CHIOU, J.-S. Investigating Effective Geometric Transformation for Image Augmentation to Improve Static Hand Gestures with a Pre-Trained Convolutional Neural Network. *Mathematics*, v. 11, n. 23, 2023. ISSN 2227-7390.

BABISHA, A.; SRIKANTH, G. U.; KIRUBA, D. A.; SUNDAR, R. Gloss-Free Sign Language Translation Using Sign2gpt-Next Technique. *2024 International Conference on Computing and Intelligent Reality Technologies (ICCIRT)*, p. 1–6, 2024.

BHARTI, S.; BALMIK, A.; NANDY, A. Novel Error Correction-Based Key Frame Extraction Technique for Dynamic Hand Gesture Recognition. *Neural Computing and Applications*, Springer Science+Business Media, v. 35, p. 21165–21180, 2023.

BHATT, R.; MALIK, K.; INDRA, G. ASL Detection in Real-Time using TensorFlow. *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, v. 2, p. 1–6, 2024.

BIANCHINI, C. S.; BORGIA, F.; MARSICO, M. D. SWift - a SignWriting Editor to Bridge between Deaf World and E-learning. *2012 IEEE 12th International Conference on Advanced Learning Technologies*, 2012.

BISHOP, C. M. *Pattern Recognition and Machine Learning*. New York: Springer, 2006. ISBN 978-0-387-31073-2.

BOUZID, Y.; JBALI, M.; GHOUL, O. E.; JEMNI, M. Towards a 3D Signing Avatar from SignWriting Notation. *Lecture Notes in Computer Science*, v. 7383 LNCS, n. PART 2, p. 229 – 236, 2012.

BOUZID, Y.; JEMNI, M. An Animated Avatar to Interpret Signwriting Transcription. *2013 International Conference on Electrical Engineering and Software Applications*, 2013.

BOUZID, Y.; JEMNI, M. TuniSigner: A Virtual Interpreter to Learn Sign Writing. *Proceedings - IEEE 14th International Conference on Advanced Learning Technologies, ICALT 2014*, p. 601 – 605, 2014.

BURIBAYEV, Z.; AOUANI, M.; ZHANGABAY, Z.; YERKOS, A.; ABDIRAZAK, Z.; ZHASSUZAK, M. Enhancing Kazakh Sign Language Recognition with BiLSTM Using YOLO Keypoints and Optical Flow. *Applied Sciences*, v. 15, p. 5685–5685, 2025.

CHEOK, M. J.; OMAR, Z.; JAWARD, M. H. A Review of Hand Gesture and Sign Language Recognition Techniques. *International Journal of Machine Learning and Cybernetics*, v. 10, p. 131–153, 2017.

CHUNG, W.-L. et al. Improving Hand Pose Recognition Using Localization and Zoom Normalizations over MediaPipe Landmarks. *Engineering Proceedings*, v. 58, 2023.

DAMDOO, R.; KUMAR, P. SignEdgeLVM Transformer Model for Enhanced Sign Language Translation on Edge Devices. *Discover Computing*, Springer Science and Business Media LLC, v. 28, 2025.

DHANALAKSHMI, P.; SREE, M.; HINDU, T.; PAPAIAH, T.; KARNA, M. Hand Gesture Recognition System using Feedback CNN. *2024 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, 2024.

ELSHAER, A. M.; AMBIOH, Y.; SOLIMAN, Z.; AHMED, O.; ELNAKIB, M.; SAFWAT, M.; ELSAYED, S.; KHALID, M. S. Enhancing Arabic Alphabet Sign Language Recognition with VGG16 Deep Learning Investigation. *2024 14th International Conference on Electrical Engineering (ICEENG)*, 2024.

FALLAH, M. K.; NAJAFI, M.; GORGIN, S.; LEE, J.-A. An Ultra-Low-Computation Model for Understanding Sign Languages. *Expert Systems with Applications*, v. 249, p. 123782, 2024. ISSN 0957-4174.

FERLIN, M.; MAJCHROWSKA, S.; NALEPA, J. Quantifying Inconsistencies in the Hamburg Sign Language Notation System. *Expert Systems with Applications*, v. 256, p. 124911, 2024.

FINK, J.; COSTER, M. D.; DAMBRE, J.; FRéNAY, B. Trends and challenges for sign language recognition with machine learning. *European Symposium on Artificial Neural Networks*, 2023.

FREITAS, F. de A.; PERES, S. M.; ALBUQUERQUE, O. de P.; FANTINATO, M. Leveraging Sign Language Processing with Formal SignWriting and Deep Learning Architectures. *Lecture Notes in Computer Science*, Springer Nature Switzerland, p. 299–314, 01 2023.

FURTADO, S. L.; OLIVEIRA, J. C. D.; SHIRMOHAMMADI, S. Interactive and Markerless Visual Recognition of Brazilian Sign Language Alphabet. *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, p. 01–06, 2023.

GANDHE, D.; MOKAR, P.; RAMANE, A.; CHOPADE, D. R. M. Sign Language Recognition for Real-time Communication. *International Journal for Research in Applied Science and Engineering Technology*, 2024.

GANGWAR, L. K.; KUMUD; BAGHELA, V. S.; TYAGI, B.; JOHRI, P. Recognition of Indian Sign Languageusing SURF, BoW CNN. *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, p. 1–7, 2024.

GOCHOO, M. *ArSL21L: Arabic Sign Language Letter Dataset*. 2022. Disponível em: <https://data.mendeley.com/datasets/f63xhm286w/1>.

GOOGLE. *MediaPipe: Framework for Building Multimodal Applied ML Pipelines*. 2020. Disponível em: <https://github.com/google-ai-edge/mediapipe>. Acesso em: 11 mar. 2025.

GULATI, N.; RAJPUT, A.; SINGH, A. Sign Language Recognition using Convolutional Neural Network. *2024 14th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, p. 876–881, 2024.

GUPTA, K.; SINGH, A.; YEDURI, S. R.; SRINIVAS, M. B.; CENKERAMADDI, L. R. Hand Gestures Recognition using Edge Computing System Based on Vision Transformer and Lightweight CNN. *Journal of ambient intelligence humanized computing*, v. 14, p. 2601–2615, 12 2022.

GüLER, O.; YüCEDAğ, Hand Gesture Recognition from 2D Images by using Convolutional Capsule Neural Networks. *Arabian Journal for Science and Engineering*, v. 47, p. 1211–1225, 2021.

HASSAN, M.; SABRI, A.; ALI, A. Detection of Arabic Sign Language by Machine Learning Techniques with PCA and LDA. *Engineering and Technology Journal*, v. 42, p. 298–311, 2024.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2. ed. [S.l.]: Springer, 2009. ISBN 978-0387848570.

HUANG, J.; CHOUVATUT, V. Video-Based Sign Language Recognition via ResNet and LSTM Network. *Imaging*, v. 10, 2024.

IMRAN, A.; RAZZAQ, A.; BAIG, I. A.; HUSSAIN, A.; SHAHID, S.; REHMAN, T. ur. Dataset of Pakistan Sign Language and Automatic Recognition of Hand Configuration of Urdu Alphabet through Machine Learning. *Data in Brief*, v. 36, p. 107021, 2021.

IOFFE, S.; SZEGEDY, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, p. 448–456, 2015.

IRVANIZAM, I.; HORATIUS, I.; SOFYAN, H. Applying Artificial Neural Network Based on Backpropagation Method for Indonesian Sign Language Recognition. *International Journal of Computing and Digital Systems*, v. 14, p. 975–985, 2023.

JIANG, Z.; MORYOSSEF, A.; MüLLER, M.; EBLING, S. Machine Translation between Spoken Languages and Signed Languages Represented in SignWriting. *Findings of the Association for Computational Linguistics: EACL 2023*, 10 2022.

JIM, A. A. J.; RAFI, I.; AKON, M. Z.; BISWAS, U.; NAHID, A.-A. KU-BdSL: Khulna University Bengali Sign Language Dataset. *Data in Brief*, 2023.

JOURNAL, I. American Sign Language (ASL) Detection System using Machine Learning. *Indian Scientific Journal Of Research In Engineering And Management*, 2023.

KAPITANOV, A.; KVANCHIANI, K.; NAGAEV, A.; KRAYNOV, R.; MAKHLIARCHUK, A. HaGRID - HAnd Gesture Recognition Image Dataset. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, p. 4572–4581, 2024.

KHATTAB, M. M.; ZEKI, A. M.; MATTER, S.; ABDELLA, M. A.; RADA; SOLIMAN, A. M. Alphabet Recognition in Arabic Sign Language: a Machine Learning Perspective. *Journal of Qena Faculty of Arts*, v. 33, p. 1–32, 2024.

KINGMA, D. P.; BA, J. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014. Disponível em: <https://arxiv.org/abs/1412.6980>.

KOLLER, O.; NEY, H.; BOWDEN, R. Deep Hand: How to Train a CNN on 1 Million Hand Images When Your Data Is Continuous and Weakly Labelled. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 06 2016.

KUMAR, C. M. N.; SUBRAMANI, A.; LAVANYA, N.; LEKHANA, N. C.; TASMIYA, R.; NISARGA, L. D. Deep Learning Based Recognition of Sign Language. *2024 Second International Conference on Data Science and Information System (ICDSIS)*, 2024.

KUMAR, N. D.; SURESH, K.; DINESH, R. CNN based Static Hand Gesture Recognition using RGB-D Data. *2022 2nd International Conference on Artificial Intelligence and Signal Processing (AISP)*, p. 1–6, 2022.

LIU, G.; SHARK, L.; HALL, G.; ZESHAN, U. Hand Motion Recognition and Visualisation for Direct Sign Writing. *2010 14th International Conference Information Visualisation*, 2010.

LOBEIRO, M.; VAZ, R. A.; ALVES, L. G. P. A Proposal to Apply SignWriting in IMSC1 Standard for the Next-Generation of Brazilian DTV Broadcasting System. *ACM International Conference Proceeding Series*, p. 230 – 233, 2022.

LOBO-NETO, V. C.; PEDRINI, H. LSWH100: a Handshape Dataset for Brazilian Sign Language (Libras) Using SignWriting. *Data in Brief*, v. 56, p. 110780–110780, 2024.

MAIA, W. F.; LOPES, A. M.; DAVID, S. A. Automatic Sign Language to Text Translation Using MediaPipe and Transformer Architectures. *Neurocomputing*, v. 642, p. 130421–130421, 2025.

MANZOOR, S.; ABBAS, Z.; CHHABRA, G.; KAUSHIK, K.; ZEHRA, M.; HAIDER, Z.; KHAN, I. U. Voice of Hearing and Speech Impaired People. *2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE)*, 05 2024.

MARQUEZ, B. Y.; ALANIS, A.; QUEZADA, A.; MAGDALENO-PALENCIA, J. S. Development of a Mobile Application with Artificial Intelligence for Mexican Sign Language Recognition. *International Journal of Interactive Mobile Technologies (iJIM)*, v. 19, p. 122–139, 2025.

MATILAINEN, M.; SANGI, P.; HOLAPPA, J.; SILVéN, O. OUHANDS Database for Hand Detection and Pose Recognition. *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, p. 1–5, 2016.

MAVI, A. *A New Dataset and Proposed Convolutional Neural Network Architecture for Classification of American Sign Language Digits*. 2021. Disponível em: <https://arxiv.org/abs/2011.08927>.

MENON, A. S.; SRUTHI, C. J.; LIJIYA, A. A 2D Fast Deep Neural Network for Static Indian Sign Language Recognition. *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, v. 1, p. 311–316, 2022.

MIAH, A. S. M.; HASAN, M. A. M.; TOMIOKA, Y.; SHIN, J. Hand Gesture Recognition for Multi-Culture Sign Language Using Graph and General Deep Learning Network. *IEEE Open Journal of the Computer Society*, v. 5, p. 144–155, 2024.

MISHRA, O.; SURYAWANSHI, P.; SINGH, Y.; DEOKAR, S. A Mediapipe-Based Hand Gesture Recognition Home Automation System. *2023 2nd International Conference on Futuristic Technologies (INCOFT)*, p. 1–6, 2023.

MOHANDES, M.; DERICHE, M.; LIU, J. A Survey on the Recognition of Arabic Sign Language. *Journal of Artificial Intelligence Review*, v. 41, p. 367–394, 2014.

NAGARAJ, A. *ASL Alphabet*. Kaggle, 2018. Disponível em: <https://www.kaggle.com/dsv/29550>.

NAIR, V.; HINTON, G. E. Rectified Linear Units Improve Restricted Boltzmann Machines. *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, p. 807–814, 2010.

NASR, M.; KADER, S. Real-Time Recognition of American Sign Language using Long-Short Term Memory Neural Network and Hand Detection. *Indonesian Journal of Electrical Engineering and Computer Science*, v. 30, p. 545, 2023.

NAVIN, N.; FARID, F. A.; RAKIN, R. Z.; TANZIM, S. S.; RAHMAN, M.; RAHMAN, S.; UDDIN, J.; KARIM, H. A. Bilingual Sign Language Recognition: a YOLOv11-Based Model for Bangla and English Alphabets. *Journal of Imaging*, v. 11, p. 134–134, 2025.

NEIVA, D. H.; ZANCHETTIN, C. Gesture recognition: A Review Focusing on Sign Language in a Mobile Context. *Expert Systems with Applications*, v. 103, p. 159–183, 2018.

NURNOBY, M. F.; EL-ALFY, E.-S. M. Multi-culture Sign Language Detection and Recognition Using Fine-tuned Convolutional Neural Network. *2023 International Conference on Smart Computing and Application (ICSCA)*, p. 1–6, 2023.

PADHI, P.; DAS, M. Hand Gesture Recognition using DenseNet201-Mediapipe Hybrid Modelling. *2022 International Conference on Automation, Computing and Renewable Systems (ICACRS)*, p. 995–999, 2022.

PASSI, P.; PEREIRA, J.; MISAL, A.; MENEZESE, V.; KIRUTHIKA, D. M. SignEase - Sign Language Interpreter Model An Indian Sign Language Interpretation Model using Machine Learning and Computer Vision Technology. *International Journal for Research in Applied Science and Engineering Technology*, 2024.

PISHARADY, P. K.; VADAKKEPAT, P.; LOH, A. P. Attention Based Detection and Recognition of Hand Postures against Complex Backgrounds. *International Journal of Computer Vision*, v. 101, p. 403–419, 2012.

PISHARADY, P. K.; VADAKKEPAT, P.; POH, L. A. Hand Posture and Face Recognition Using Fuzzy-Rough Approach. *Computational Intelligence in Multi-Feature Visual Pattern Recognition: Hand Posture and Face Recognition using Biologically Inspired Approaches*, Springer Singapore, Singapore, p. 63–80, 2014.

POORNIMA, B. V.; SRINATH, S. Indian Sign Language Recognition using CNN with Spatial Pyramid and Global Average Pooling. *Indian journal of computer science and engineering*, v. 15, n. 2, p. 227–238, 2024.

PRIEUR, J.; BARBU, S.; BLOIS-HEULIN, C.; LEMASSON, A. The Origins of Gestures and Language: History, Current Advances and Proposed Theories. *Biological Reviews*, v. 95, n. 3, p. 531–554, 2020.

PRILLWITZ, S.; LEVEN, R.; ZIENERT, H.; HANKE, T.; HENNING, J. HamNoSys Version 2.0: Hamburg Notation System for Sign Languages: An Introductory Guide. *International Studies on Sign Language and Communication of the Deaf*, Signum Press, v. 5, 1989.

PUGEAULT, N.; BOWDEN, R. Spelling It out: Real-time ASL Fingerspelling Recognition. *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 11 2011.

PURI, M.; ARORA, S.; ROHAN; VINAYAK; NAGRATH, P. Breaking the Silence: Empowering the Deaf, Mute Blind Community through Sign Language Live Captioning Using Deep Learning. *2023 3rd Asian Conference on Innovation in Technology (ASIANCON)*, p. 1–8, 2023.

QUIROGA, F. M.; ANTONIO, R.; RONCHETTI, F.; LANZARINI, L.; ROSETE, A. A Study of Convolutional Architectures for Handshape Recognition applied to Sign Language. *SEDICI - Repositorio de la Universidad Nacional de La Plata*, 2017.

RAFI, A. M.; NAWAL, N.; BAYEV, N. S. N.; NIMA, L.; SHAHNAZ, C.; FATTAH, S. A. Image-based Bengali Sign Language Alphabet Recognition for Deaf and Dumb Community. *2019 IEEE Global Humanitarian Technology Conference (GHTC)*, p. 1–7, 2019.

RANGU, N.; DATHI, P.; KETHAVATH, P.; MANNE, S.; JAMAL, K. Sign Language Recognition using Transfer Learning. *2024 4th International Conference on Intelligent Technologies (CONIT)*, p. 1–6, 2024.

RENJITH, S.; SURESH, M. S.; RASHMI, M. An Effective skeleton-based Approach for Multilingual Sign Language Recognition. *Engineering Applications of Artificial Intelligence*, v. 143, p. 109995–109995, 2025.

ROBERT, E. J.; DURAISAMY, H. J. A Review on Computational Methods Based Automated Sign Language Recognition System for Hearing and Speech Impaired Community. *Concurrency and Computation: Practice and Experience*, 2023.

RODRIGUEZ, M.; OUBRAM, O.; BASSAM, A.; LAKOUARI, N.; TARIQ, R. Mexican Sign Language Recognition: Dataset Creation and Performance Evaluation Using MediaPipe and Machine Learning Techniques. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 14, p. 1423–1423, 2025.

RONCHETTI, F.; QUIROGA, F.; JEREMIAS, U.; RIOS, G. G.; BIANCO, P. D.; HASPERUé, W.; LANZARINI, L. Comparison of Small Sample Methods for Handshape Recognition. *Journal of Computer Science and Technology*, v. 23, p. e03–e03, 2023.

RONCHETTI, F.; QUIROGA, F.; LANZARINI, L.; ESTREBOU, C. Handshape Recognition for Argentinian Sign Language using ProbSom. *Journal of Computer Science and Technology*, v. 16, n. 1, p. 1–5, 2016. ISSN 1666-6038.

SABATO, M.; SANDRONI, S.; MARCECA, M. Deafness and Communicative Barriers with Reference to University Courses in the Rehabilitation Area. *European journal of public health*, Oxford University Press, v. 33, 2023.

SAHOO, J. P.; SAHOO, S. P.; ARI, S.; PATRA, S. K. RBI-2RCNN: Residual Block Intensity Feature Using a Two-stage Residual Convolutional Neural Network for Static Hand Gesture Recognition. *Signal, Image and Video Processing*, 02 2022.

SELVARAJ, P.; GOKUL, N. C.; KUMAR, P.; KHAPRA, M. M. OpenHands: Making Sign Language Recognition Accessible with Pose-based Pretrained Models across Languages. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022.

SEVILLA, A. F. G.; ESTEBAN, A. D.; LAHOZ-BENGOECHEA, J. M. Automatic SignWriting Recognition: Combining Machine Learning and Expert Knowledge to Solve a Novel Problem. *IEEE Access*, Institute of Electrical and Electronics Engineers, v. 11, p. 13211–13222, 2023.

SHIRUDE, R.; KHURANA, S.; SREEMA, R.; TURUK, M.; JAGDALE, J.; CHIDREWAR, S. *Indian Sign Language Recognition System using GAN and Ensemble Based Approach*. 2024.

SIDHU, M.; HON, S. P.; MARATHE, S.; RANE, T. Convolutional Neural Networks for Indian Sign Language Recognition. *International journal of innovative science and research technology*, p. 2568–2573, 2024.

SINDHU, K. S.; MEHNAAZ; NIKITHA, B.; VARMA, P. L.; UDDAGIRI, C. Sign Language Recognition and Translation Systems for Enhanced Communication for the Hearing Impaired. *2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU)*, p. 1–6, 2024.

SONAWANE, V. *Indian Sign Language Dataset*. 2020. Disponível em: <https://www.kaggle.com/datasets/vaishnaviasonawane/indian-sign-language-dataset/data>.

SRIKANTARAO, S.; VOSURI, N.; EDAGOTTI, J.; RAHAMAN, S.; ASHARF, S. M. Indian Sign Language Detection Using Deep Learning. *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*, 2024.

SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research*, v. 15, n. 1, p. 1929–1958, 2014.

STIEHL, D.; ADDAMS, L.; OLIVEIRA, L. E.; GUIMARAES, C.; BRITTO, A. S. Towards a SignWriting Recognition System. *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, 2015.

SURJO, G. S.; GHOSH, B. K.; ALAM, M. J.; MAHAMUDULLAH; RAZIB, M.; BILGAIYAN, S. A Comparative Analysis Between Single & Dual-Handed Bangladeshi Sign Language Detection using CNN based Approach. *2023 International Conference on Computer Communication and Informatics (ICCCI)*, p. 1–8, 2023.

SUTTON, V. *SignWriting for Sign Languages*. 1974. Disponível em: <https://www.signwriting.org/>.

SUTTON, V. SignWriting for Sign Languages. SignWriting Organization, 1974.

TAKYI, K.; MENSAH, O.; GUEUWOU, S. M.; NYARKO, M. S.; ADADE, R.; BORKOR, R. N.; BOADU-ACHEAMPONG, S. I.; TABARI, L. AfriSign: African Sign Languages Machine Translation. *Discover Artificial Intelligence*, Springer Nature, v. 5, 2025.

TURNER, M.; SMITH, A. Robustness in Hand Gesture Recognition with MediaPipe Landmarks: An Empirical Study. *ArXiv Preprint*, 2023. Disponível em: <https://arxiv.org/pdf/2305.05296v1>.

WAGHMARE, e. a. P. P. A Study on Techniques and Challenges in Sign Language Translation. *International Journal on Recent and Innovation Trends in Computing and Communication*, v. 11, p. 4039–4052, 2023.

WOLFF, M. M.; ANDERSON, C.; BANIć, A. Towards Integrating American Sign Language into Virtual Reality. *Proceedings - 2024 IEEE International Symposium on Mixed and Augmented Reality Adjunct, ISMAR-Adjunct 2024*, p. 415 – 416, 2024.

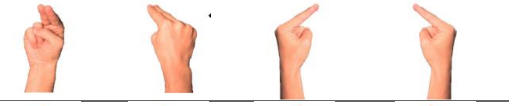
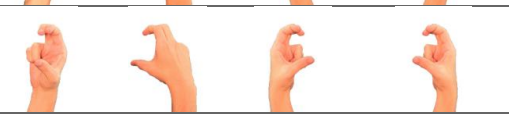
YAZDANI, S.; GENABITH, J. V.; ESPAñA-BONET, C. Continual learning in multilingual sign language translation. *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics*, p. 10923–10938, 2025.

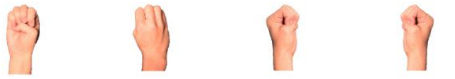
YEWARE, R.; GOKHALE, A.; CHALSE, S.; SAMDEKAR, M.; MANTE, J. American Sign Language Detection using Deep Learning. *2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, p. 1–6, 2023.

ZHANG, Y.; WANG, J.; WANG, X.; JING, H.; SUN, Z.; CAI, Y. Static Hand Gesture Recognition Method Based on the Vision Transformer. *Multimedia Tools and Applications*, v. 82, p. 31309–31328, 2023.

ANEXO A – LISTAGEM DE GESTOS DO SIGNWRITING

Code	Name	Front	Back	Left	Right
S100	Index				
S101	Index on Circle				
S103	Index on Oval				
S105	Index on Angle				
S106	Index Bent				
S107	Index Bent on Circle				
S10a	Index Cup				
S10b	Index Hinge on Fist				
S10c	Index Hinge on Fist Low				
S10e	Index Middle				
S110	Index Middle Bent				
S112	Index Middle Hinge				
S115	Index Middle Unit				

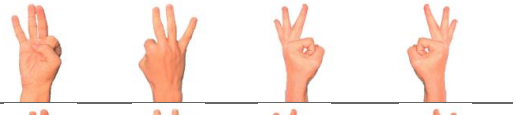
S118	Index Middle Unit Cup	
S119	Index Middle Unit Hinge	
S11a	Index Middle Cross	
S11c	Middle Bent Over Index	
S11e	Index Middle Thumb	
S124	Index Up, Middle Hinge, Thumb Side	
S127	Index Middle Up Spread, Thumb Forward	
S128	Index Middle Thumb Cup	
S12a	Index Middle Thumb Hook	
S12b	Index Middle Thumb Hinge	
S12d	Index Middle Unit, Thumb Side	
S12e	Index Middle Unit, Thumb Tight	
S133	Index Middle Cross, Thumb Side	
S135	Index Middle Unit Cup, Thumb Forward	

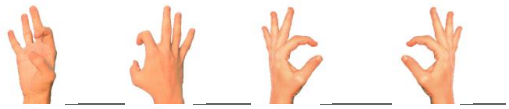
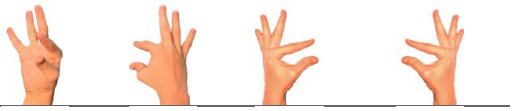
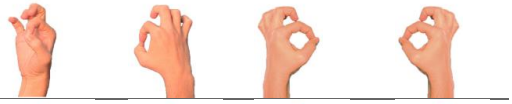
S13d	Index Middle Thumb, Unit Hinge	
S13f	Index Middle Thumb Angle	
S140	Middle Thumb Angle Out, Index Up	
S142	Middle Thumb Angle, Index Up	
S144	Four Fingers	
S147	Four Fingers Unit	
S14a	Four Fingers Unit Bent	
S14c	Five Fingers Spread	
S14e	Five Fingers Spread, Four Bent	
S150	Five Fingers Spread, All Bent	
S151	FiveFingers Spread, All Bent Heel	
S152	Five Fingers Spread, Thumb Forward	
S153	Five Fingers Spread Cup	
S154	Five Fingers Spread Cup Open	

S155	Five Fingers Spread Hinge Open	
S157	Five Fingers Spread Hinge	
S15a	Flat Hand	
S15d	Flat Thumb Side	
S160	Flat Thumb Forward	
S168	Claw No Thumb	
S169	Claw Thumb Forward	
S16c	Open Cup	
S16d	Cup	
S16f	Cup Thumb Side	
S170	Open Cup No Thumb	
S171	Cup No Thumb	
S172	Open Cup Thumb Forward	
S173	Cup Thumb Forward	

S174	Open Curlicue	
S175	Curlicue	
S176	Circle	
S177	Oval	
S178	Oval Thumb Side	
S17b	Open Hinge	
S17c	Open Hinge Thumb Forward	
S17d	Hinge	
S17e	Small Hinge	
S17f	Open Hinge Thumb Side	
S180	Hinge Thumb Side	
S181	Open Hinge No Thumb	
S182	Hinge No Thumb	
S185	Angle	

S186	Index Middle Ring	
S187	Index Middle Ring on Circle	
S18b	Index Middle Ring, Bent	
S18c	Index Middle Ring, Unit	
S18d	Index Middle Ring, Unit Hinge	
S192	Baby Up	
S193	Baby Up On Fist Thumb Under	
S194	Baby Up On Circle	
S195	Baby Up On Oval	
S198	Baby Bent	
S19a	Baby Thumb	
S19c	Baby Index Thumb	
S1a0	Baby Index	

S1a1	Baby Index on Circle	
S1a3	Baby Index on Angle	
S1a4	Index Middle Baby	
S1a5	Index Middle Baby on Circle	
S1a7	Ring Hinge	
S1a8	Index Middle Baby on Angle	
S1b1	Ring Baby on Circle	
S1b2	Ring Baby on Oval	
S1bb	Index Ring Baby on Circle	
S1c1	Index Ring Baby on Hinge	
S1c3	Index Ring Baby on Angle	
S1c5	Middle Hinge	
S1cd	Middle Ring Baby	
S1ce	Middle Ring Baby on Circle	

S1d0	Middle Ring Baby on Cup	
S1d1	Middle Ring Baby on Hinge	
S1d2	Middle Ring Baby on Angle Out	
S1d3	Middle Ring Baby on Angle In	
S1d4	Middle Ring Baby on Angle	
S1d5	Middle Ring Baby Bent	
S1d7	Middle Ring Baby Unit on Claw Side	
S1d8	Middle Ring Baby Unit on Hook Out	
S1da	Middle Ring Baby Unit on Hook	
S1dc	Index Thumb Side	
S1de	Index Thumb Side, Thumb Diagonal	
S1df	Index Thumb Side, Thumb Unit	
S1e1	Index Thumb Side, Index Bent	
S1e4	Index Thumb Forward, Index Straight	

S1e8	Index Thumb Curve, Thumb Side	
S1ea	Index Thumb Curve, Thumb Under	
S1eb	Index Thumb Circle	
S1ec	Index Thumb Cup	
S1ed	Index Thumb Cup Open	
S1ee	Index Thumb Hinge Open	
S1ef	Index Thumb Hinge Large	
S1f0	Index Thumb Hinge	
S1f1	Index Thumb Hinge Small	
S1f2	Index Thumb Angle Out	
S1f4	Index Thumb Angle	
S1f5	Thumb	
S1f7	Thumb Side Diagonal	
S1f8	Thumb Side Unit	

S1f9	Thumb Side Bent	
S1fa	Thumb Forward	
S1fb	Thumb Between Index Middle	
S1fc	Thumb Between Middle Ring	
S1fd	Thumb Between Ring Baby	
S1ff	Thumb Over Two Fingers	
S201	Thumb Under Four Fingers	
S203	Fist