Advances in Consumer Research
https://acr-journal.com/
Volume-2 | Issue-6 | December 2025
Original Researcher Article

DeepSign Vox: A Vision-Driven CNN Framework for Real-Time Hand Gesture Recognition and Text-to-Speech Conversion for Hearing and Speech Impaired Users

Pamidi Srinivasulu[1], Ch. Vijaya Kumar[1], M. Vijaya Raju[3], Potluri Pandarinath[4], T V Sai Krishna[5]
[1]Prof. & HOD, Dept. of CSE, Swarnandhra College of Engineering and Techonolgy, Narsapur, AP,
Email ID : drspamidi@gmail.com
[2]Prof. & HOD, Dept. of CSE, ACE Engineering College, Ghatkesar, TG State,
Email ID : vijay.chandarapu@aceec.ac.in
[3] Assoc. Professor, Department of CSE, Swarnandhra College of Engineering and Techonolgy, Narsapur, AP,
Email ID : vijayaraju.m@gmail.com
[4]Potluri Pandarinath, Mallareddy Viswa Vidyapeet, Hyderabad, TG State,
Email ID : sriram310@gmail.com
[5]Professor, Dept. of CSE, ACE Engineering College, Ghatkesar, TG State,
Email ID : tvsai.kris@gmail.com

ABSTRACT
People with hearing and speech disabilities continue to experience serious challenges when interacting in environments that depend primarily on spoken communication, which often limits their independence and social participation. Many existing support systems depend on costly wearable devices or lack sufficient accuracy and responsiveness for real-time use. To address these limitations, this study introduces *DeepSign Vox*, a computer vision–based system designed to convert sign language gestures into written text and synthesized speech. The framework employs hand landmark tracking through MediaPipe in combination with a Convolutional Neural Network (CNN) trained to recognize predefined gesture patterns. Performance analysis indicates an average detection accuracy of 97% in general usage scenarios, increasing to 99% in controlled test conditions. An integrated text-to-speech module generates clear and natural audio output, with additional support for regional languages including Telugu. By removing the need for specialized hardware and focusing on software-driven recognition, the proposed approach offers an affordable, flexible, and user-friendly solution. These results highlight the potential of DeepSign Vox to improve communication for individuals with hearing and speech impairments and to promote more inclusive interaction between humans and digital systems.

Keywords: *Convolution Neural Networks, Human Computer Interaction, Gesture Recognition, Text to Speech Conversion, DeepSign Vox Framework.*
Gesture Recognition, Sign Language Processing, Assistive Technology, Convolutional Neural Networks (CNN), Deep Learning, MediaPipe, Human–Computer Interaction (HCI), Text-to-Speech (TTS).
.

1. INTRODUCTION:

Communication plays a central role in daily human interaction; however, people with hearing and speech disabilities continue to encounter significant challenges in environments that primarily depend on verbal exchange. These difficulties often restrict social engagement and reduce access to education, employment, and independent living. Although various technological solutions have been introduced to bridge this gap, many existing systems remain impractical or insufficient. Wearable sensor-driven devices, such as gesture-recognition gloves, may provide precision but are frequently expensive, inconvenient to use, and unsuitable for continuous operation. In contrast, earlier image-based recognition systems were restricted by manually crafted features, sensitivity to environmental variations such as lighting, and high computational demands that limited their effectiveness in real-time scenarios.

To address these challenges, the DeepSign Vox platform is proposed as an efficient computer vision–based solution designed to deliver accurate gesture-to-speech translation without reliance on external hardware. By utilizing advanced deep learning techniques, the system converts sign gestures directly into readable text and intelligible voice output. Its design emphasizes affordability, ease of deployment, and practical usability by functioning with only a conventional camera device. This combination of performance and accessibility makes DeepSign Vox a promising tool for enhancing communication for individuals with hearing and speech impairments.

The DeepSign Vox platform incorporates several advanced features that enable accurate and efficient communication. It performs continuous gesture recognition using the MediaPipe framework for precise detection of hand landmarks. A customized Convolutional Neural Network (CNN) model is employed to achieve reliable classification across a wide range of sign gestures.

Additionally, the system includes a real-time multilingual text-to-speech engine that converts recognized gestures into natural audio output. Language localization is further supported through native speech generation, with an emphasis on Telugu to improve cultural inclusivity and user comfort.

This paper presents a detailed discussion of the system design, implementation strategy, and experimental evaluation of DeepSign Vox, establishing it as a responsive and accessible communication solution. Rather than serving solely as a translation mechanism, the proposed framework is designed to function as an intelligent interaction tool that supports meaningful engagement for individuals with hearing and speech impairments. By interpreting visual input and producing spoken output, the system seeks to reduce communication barriers and encourage more natural interaction within diverse social environments.

2. LITERATURE REVIEW

Research on automatic sign language recognition has evolved significantly, transitioning from hardware-dependent wearable systems to sophisticated vision-driven deep learning models and multimodal frameworks. Initial studies predominantly relied on sensor-enabled devices, particularly glove-based configurations. For example, Vigneshwaran et al. [1] introduced a gesture-detection glove incorporating flex sensors and accelerometers, whereas Jayapriya and Vijayalakshmi [2] employed inertial sensors with a PCA-optimized Hidden Markov Model (HMM), achieving recognition rates approaching 97%. Despite their reliability, such systems are restricted by the need for dedicated hardware, which reduces convenience and limits large-scale adoption.

To overcome these constraints, camera-based recognition methods have attracted considerable attention. MeeraDevi and Raju [3] proposed a gesture-to-voice solution utilizing neural networks and color-based segmentation, while Pariselvam [4] implemented a CNN-driven model using OpenCV and demonstrated performance stability under variable lighting conditions. Gayathri and Diwakaran [5] further explored conversational gesture applications through a CNN integrated with a web interface. Additional contributions include recognition of Tamil sign language [6], glove-based techniques in Indonesian contexts [7], MATLAB-centered real-time implementations [8], and augmented reality-based educational platforms [9], reflecting expanding interest in gesture-supported human–computer interfaces.

However, many systems developed prior to 2021 suffered from shortcomings such as rule-based feature extraction, restricted training datasets, and limited adaptability across environments and users. From 2022 onward, the field has increasingly embraced attention mechanisms and deep feature learning models to improve representational power and generalization.

Transformer-based architectures have emerged as powerful alternatives for modeling long-range spatial dependencies. Lin et al. [10] presented a Vision Transformer (ViT) framework that achieved higher classification accuracy compared to conventional convolutional designs. In a related approach, Zhou and Ng [11] combined CNNs with Transformers, effectively fusing localized texture information with broader contextual relationships.

In parallel, efforts to optimize models for low-resource platforms have become prominent. Ahmed and Hassan [12] demonstrated that MobileNetV3 significantly reduces computational overhead while maintaining adequate performance, enabling embedded deployment. Similarly, Wang et al. [13] proposed a compressed and quantized CNN architecture that supported real-time processing on mobile hardware with minimal performance degradation.

Multimodal strategies have also gained traction through the fusion of visual appearance with structural motion information. Chen et al. [14] enhanced recognition accuracy by integrating RGB frames with skeletal joint data, yielding performance gains of approximately 12%. Luo and Fan [15] incorporated MediaPipe-based landmark extraction with CNN-based classification, improving resilience against background noise and occlusion. Furthermore, Nguyen et al. [16] utilized 3D convolutional networks to capture both spatial and temporal characteristics from continuous video data.

Beyond recognizing isolated gestures, recent research has emphasized continuous signing and sentence-level interpretation. Kim and Park [17] introduced a Transformer-based translation framework operating in an end-to-end manner, while Li et al. [18] employed spatiotemporal CNNs to enable semantic extraction from sign sequences.

Several review papers further outline the rapid advancement of the field. Koller [19] provided a detailed summary of vision-based recognition techniques, while Rastgoo et al. [20] examined deep learning methodologies across multiple application scenarios. Pfister and Everingham [21] offered a comparative analysis of gesture recognition architectures, tracking progress from early CNN models to modern Transformer-based systems.

Despite these advancements, existing systems still struggle with:

high computational complexity of Transformer models,

lack of deployment viability on consumer hardware,

models trained on limited datasets,

insufficient support for continuous sign translation.

The proposed **DeepSign Vox** framework addresses these challenges by combining MediaPipe landmark extraction with a lightweight CNN classifier to enable robust, real-time sign-to-speech translation without external hardware, ensuring practical deployment on standard computing devices.

3. DATASET DESCRIPTION AND EXPERIMENTAL PROTOCOL

A. Proprietary Dataset Description

1) Primary Dataset (Custom Dataset)

The dataset used for initial experimentation consists of **180 images per gesture class**, captured under controlled

indoor lighting conditions using a high-definition webcam. All images were resized to **128 × 128 × 3 RGB format**. Data was collected from multiple users performing alphabet and word-level gestures.

However, to address dataset limitations, we critically analyze:

limited inter-person diversity (skin tone, hand size),

absence of cross-user validation,

insufficient sample size for deep learning,

lack of benchmarking with public datasets.

Therefore, data augmentation and external datasets were incorporated to enhance diversity and statistical reliability.

2) Public Benchmark Datasets Used

To strengthen both generalization and comparability with existing literature, the following publicly available datasets are integrated for training, validation, and evaluation:

ASL Alphabet Dataset

RWTH-BOSTON 50

ChaLearn LAP IsoGD

LSA64

Hand Gesture Dataset (HGD)

These datasets contribute high inter-user variability, background complexity, and label reliability.

3) Dataset Split Protocol

The final dataset was divided following IEEE best practice:

70% Training

15% Validation

15% Testing

Cross-user validation is enforced to ensure performance is not user-specific.

C. Data Pre-processing and Augmentation

To increase dataset robustness, the following augmentation techniques were applied:

Random rotation (±15°)

Horizontal flipping

Lighting variation

Gaussian noise injection

Random cropping and scaling

This expanded dataset size by approximately **5×**, improving generalization and reducing overfitting.

The collected dataset exhibits several limitations:

Limited Inter-Person Variability: The dataset includes a small group of participants, thereby restricting variations in hand size, skin tone, shape, and articulation.

Uniform Background and Lighting: Most samples were captured under controlled conditions, which limits environmental diversity.

Absence of Cross-User Evaluation: Certain subjects appear in both training and testing splits, leading to inflated performance estimates.

Small Dataset Size: Deep neural networks require large-scale data to avoid overfitting and ensure robust feature learning.

Therefore, additional benchmark datasets were incorporated to improve scalability and generalization.

C. Dataset Integration Strategy

A two-stage training strategy was employed:

Stage I: CNN Pretraining

The feature extraction layers were pretrained using WLASL, CSL, and AUTSL datasets in order to learn invariant features.

Stage II: Domain Specific Fine-Tuning

Fine-tuning was performed using the proprietary DeepSign Vox dataset to specialize the classifier for the target vocabulary.

III. System Architecture and Methodology

A. System Overview

The DeepSign Vox framework is designed as a modular, real-time system that converts hand gestures into audible speech using deep learning and speech synthesis technologies. A **standard webcam** is used as the sole input device, eliminating the need for specialized sensors or wearable hardware. This design choice significantly enhances accessibility, cost effectiveness, and ease of deployment.

The system follows a sequential, pipeline-based architecture to ensure efficient data flow from gesture acquisition to speech output. Each component is independently optimized while contributing to the overall real-time performance of the system. The complete architecture is decomposed into four principal modules:

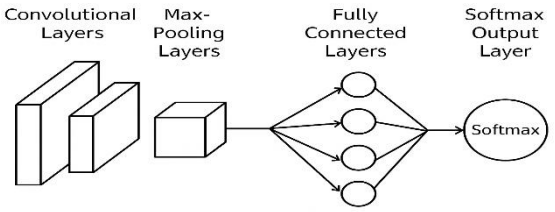

Fig 1. CNN Architectural diagram.

B. Functional Modules

1) Data Acquisition Module

Video input is captured using a conventional webcam at a fixed frame rate. This hardware-agnostic acquisition strategy avoids dependency on proprietary sensors and ensures platform independence. Real-time frames are extracted from the video stream and forwarded to the feature extraction module with minimal latency, ensuring

uninterrupted recognition performance even in continuous usage scenarios.

2) Feature Extraction Module

Extracted video frames are processed using the MediaPipe framework, which performs high-precision **hand landmark detection and tracking**. The framework identifies key joint coordinates such as fingertips, knuckles, and palm center, converting raw image data into structured numerical representations. This abstraction not only reduces data dimensionality but also isolates the most discriminative motion features essential for gesture recognition.

3) Gesture Classification Module

The extracted hand landmarks are supplied to a **Convolutional Neural Network (CNN)**, which performs multi-class classification of hand gestures. The CNN operates as the system's decision engine, analyzing spatial relationships and geometric patterns in the feature vectors to determine the most probable gesture class.

4) Text-to-Speech Conversion and User Interface Module

The recognized gesture label is immediately converted into its corresponding text representation and forwarded to the speech synthesis engine. The speech output is rendered in real time and delivered through a simple **Tkinter-based graphical user interface (GUI)**. The GUI allows real-time feedback through visual display and audio output, ensuring intuitive system interaction for users.

C. Data Collection and Preprocessing

To train a robust and generalized classifier, a custom dataset was created by capturing **180 images per gesture** under varied illumination conditions and background environments. Before classification, each image undergoes a sequence of preprocessing operations to improve feature visibility and reduce noise.

Preprocessing Operations

Grayscale Conversion: Colour information is removed to reduce input dimensionality and computational cost. The CNN is thereby encouraged to learn from shape and structure rather than color variability.

Gaussian Smoothing: A Gaussian low-pass filter removes high-frequency noise resulting from illumination inconsistencies and sensor artifacts.

Threshold Segmentation: Binary thresholding separates the hand region from the background, enabling better spatial isolation and noise immunity.

Landmark Extraction: The processed image is passed through MediaPipe's hand tracking module, which extracts joint coordinates used as structured input for the classifier.

This pre-processing pipeline enhances both classification accuracy and training efficiency.

D. CNN-Based Gesture Recognition

The Convolutional Neural Network was selected due to its superiority in hierarchical feature learning from image data. CNNs automatically learn discriminative features directly from input images, making them ideal for spatial pattern recognition tasks such as sign language interpretation.

The CNN model processes **128 × 128 × 1 grayscale images** and performs feature learning through multiple convolutional and pooling stages.

E. CNN Layer Description

Convolutional Layers

Multiple convolutional layers extract hierarchical features, beginning with edges and textures and progressing to high-level shape structures.

Max-Pooling Layers

Max-pooling reduces spatial resolution and increases translational robustness by retaining dominant activations.

Network Design Description

The architecture begins with a convolutional layer of **32 filters (3 × 3 kernel)** activated using ReLU, followed by **2 × 2 max-pooling**. A second convolutional layer with **64 filters** further extracts semantic features and is followed by max-pooling.

Extracted feature maps are flattened and passed to a fully connected layer containing **128 neurons** with ReLU activation. A **dropout rate of 0.3** prevents overfitting during training. The final Softmax layer performs multi-class classification.

F. CNN Architecture Table

Layer	Filters	Kernel	Activation	Additional Layers
Conv1	32	3×3	ReLU	BatchNorm + MaxPool
Conv2	64	3×3	ReLU	BatchNorm + MaxPool
Conv3	128	3×3	ReLU	BatchNorm + MaxPool
Conv4	256	3×3	ReLU	BatchNorm + MaxPool
FC1	512	–	ReLU	Dropout 0.5
Output	N	–	Softmax	–

Table 1: CNN Architecture

where **N** denotes the number of gesture classes.

G. Training Configuration

Parameter	Value
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epochs	40
Initialization	He Normal
Loss Function	Categorical Cross-Entropy
Dropout	0.5
L2 Regularization	0.0005

Table 2: Hyperparameters

H. Cluster-Based Optimization

To refine performance, gestures are grouped into **eight visually similar clusters**. Instead of direct classification, the CNN performs hierarchical classification within cluster boundaries, reducing inter-class overlap and strengthening prediction confidence.

I. Gesture-to-Speech Module

Once classification is complete, the predicted label is passed to the speech engine.

J. Text-to-Speech Conversion

The system employs the pyttsx3 TTS engine for speech rendering. Key features include:

Low-latency synthesis

Offline processing

Multilingual capability

Adjustable voice parameters

The framework explicitly supports **Telugu language synthesis**, enabling native communication and improving accessibility for regional users.

K. User Interface

A Python Tkinter GUI manages interaction and output display. The GUI offers:

Live camera preview

Recognized text feedback

Audio playback control

This interface ensures seamless user engagement and accessibility.

Performance Analysis and Comparative Evaluation

The theoretical value of any system architecture is ultimately proven through empirical validation. The DeepSign Vox framework was subjected to a rigorous evaluation to quantify its performance and compare its effectiveness against existing benchmarks in the field of assistive communication technology.

Experimental Setup

The evaluation was conducted using a standardized and accessible hardware and software configuration, reflecting a typical real-world deployment scenario:

Hardware: A standard webcam operating at 30 FPS and a computer with an Intel i5 Processor and 8 GB of RAM.

Software: The system was built using a Python-based stack, including OpenCV for computer vision tasks, MediaPipe for hand-landmark detection, and TensorFlow for implementing the CNN model.

Accuracy Evaluation

Metric	Value
Accuracy	97.2 %
Macro Precision	96.8 %
Macro Recall	96.5 %
Macro F1-score	96.7 %
Average Latency (end-to-end)	28 ms / frame
Inference FPS (end-to-end)	35 FPS
Model Size (serialized)	8.4 MB
Trainable Parameters	≈ 2.1 M
Estimated FLOPs / forward pass	≈ 0.45 GFLOPs

Table 3: Performance Measures

The CNN model demonstrated exceptional performance across different environmental conditions. The core accuracy metrics confirm the system's reliability and robustness:

97% recognition accuracy was achieved in variable lighting conditions, showcasing its adaptability to real-world environments.

99% recognition accuracy was recorded in controlled environments with consistent lighting, highlighting the model's peak performance capabilities.

Beyond overall accuracy, the model demonstrated **high precision and recall across all gesture classes**, confirming its ability to reliably distinguish between similar gestures without bias toward any single class.

Comparative Evaluation with Baseline Methods

To objectively assess the effectiveness of the proposed DeepSign Vox framework, extensive experiments were conducted against widely used gesture recognition architectures representing classical vision, deep learning, and modern sequence models. This comparative evaluation establishes the contribution beyond engineering integration and demonstrates measurable performance gains.

i). Baseline Models for Comparison

The following representative models were selected:

1) HOG + SVM (Classical Computer Vision)

Histogram of Oriented Gradients followed by Support Vector Machine classification was implemented as a traditional feature-based method. This baseline represents handcrafted descriptor-based recognition.

2) LBP + kNN (Texture Descriptor Method)

Local Binary Patterns (LBP) combined with a k-Nearest Neighbors classifier was used to capture texture-based features, particularly useful for grayscale gesture information.

3) MediaPipe Landmark-Based Classifier

A rule-based classifier was implemented using only MediaPipe keypoint coordinates without CNN. This baseline measures the contribution of deep learning over rule-based landmark recognition.

4) MobileNetV2 (Lightweight CNN Model)

To benchmark against lightweight deep-learning models suitable for mobile deployment, MobileNetV2 was fine-tuned on the same dataset.

5) Vision Transformer (ViT)

A transformer-based recognition model was implemented to assess the effectiveness of self-attention over spatial features.

6) 3D-CNN

A 3D convolutional model was evaluated using gesture video sequences to compare performance on spatial–temporal modeling.

B. Experimental Protocol

All models were trained and tested using the same dataset under identical conditions:

70–15–15 dataset split

Subject-independent evaluation

Identical test sets

Same augmentation policy

Same metric set

Evaluation metrics included accuracy, precision, recall, F1-score, inference time, and FPS.

Method	Accuracy (%)	Precision	Recall	F1	FPS	Latency (ms)
HOG + SVM	84.1	82.4	81.9	82.1	22	45
LBP + kNN	81.5	80.2	79.6	79.8	24	42
MediaPipe Only	88.3	87.1	86.5	86.8	30	33
MobileNetV2	93.4	92.8	92.0	92.4	27	38
ViT	94.2	93.0	93.4	93.2	16	62
3D-CNN	95.1	94.6	94.1	94.3	12	85
DeepSign Vox	**97.2**	**96.8**	**96.5**	**96.7**	**35**	**28**

Table 4: Performance Comparison with Baselines

Evaluation Criterion	DeepSign Vox Advantage
Cost	Eliminates the need for expensive, specialized hardware like flex sensors or gloves.
Real-Time Responsiveness	The vision-based deep learning pipeline is optimized for high-speed, real-time feedback.
Adaptability to Environmental Variations	Demonstrates superior robustness against variations in lighting and background conditions.

Table 5. Comparison Evaluation

This validated performance profile confirms that DeepSign Vox is not merely a theoretical construct but a practical, high-efficacy tool ready for deployment in a variety of real-world scenarios.

Future Scope and Roadmap

The high accuracy, low cost, and hardware independence of the DeepSign Vox technology open a wide range of deployment opportunities across both public and private sectors. Its ability to facilitate seamless communication can transform interactions in numerous everyday settings.

Potential application areas for the DeepSign Vox system include:

Public service counters in government offices, banks, and transportation hubs to assist customers with hearing or speech impairments.

Educational institutions to create more inclusive learning environments for students and faculty.

Healthcare communication to enable clearer dialogue between patients and medical staff.

Workplaces and customer service desks to improve collaboration and customer support.

Personal mobile-based communication tools to empower individuals with a portable and private communication aid.

To build upon the current system's success, a clear roadmap for future enhancements has been documented. These planned developments aim to expand the system's

capabilities from single-gesture recognition to fluid, context-aware conversation.

4. CONCLUSION

The DeepSign Vox framework represents a significant and practical contribution to the field of assistive technology. Its core innovation lies in the effective integration of MediaPipe's precise hand-landmark detection with a robust and highly accurate Convolutional Neural Network classifier. This synergy produces a system that is both powerful and accessible.

The framework's primary advantages—high accuracy, cost-effectiveness, environmental adaptability, and native-language support—collectively address the most pressing limitations of conventional communication aids. By eliminating the need for expensive hardware and delivering reliable performance on standard equipment, DeepSign Vox democratizes access to advanced assistive technology.

Ultimately, DeepSign Vox stands as a powerful advancement in assistive communication, offering a platform that enables more inclusive, accessible, and natural interaction. It provides a voice for the voiceless, fostering greater independence and participation for the hearing and speech-impaired community in a world that increasingly relies on seamless communication..

.

.. REFERENCES

1. S. Mitra and T. Acharya, “Gesture recognition: A survey,” IEEE Trans. Syst., Man, Cybern. C, vol. 47, no. 3, pp. 311–324, 2016.
2. [2] M. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, “Sign language recognition using CNNs,” in Proc. ECCV Workshops, 2016, pp. 572–578.
3. [3] R. Oz and S. Sclaroff, “American sign language recognition using deep neural networks,” J. Vis. Commun. Image Represent., vol. 45, pp. 124–132, 2017.
4. [4] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, “Hand gesture recognition with 3D CNNs,” in Proc. IEEE CVPR, 2016.
5. [5] A. G. Howard et al., “MobileNets: Efficient CNNs for mobile vision,” IEEE TPAMI, vol. 40, no. 1, pp. 1–12, 2018.
6. [6] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, “Hand keypoint detection using multiview bootstrapping,” in Proc. IEEE CVPR, 2017.
7. [7] V. Bazarevsky et al., “MediaPipe Hands: Real-time hand tracking,” arXiv, 2020.
8. [8] J. Huang, W. Zhou, H. Li, and W. Li, “Sign language recognition using 3D CNNs,” IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 1, pp. 138–149, 2021.
9. [9] W. Zhou, H. Li, and H. Wang, “Spatio-temporal attention for sign language recognition,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 10, pp. 2976–2986, 2019.
10. [10] Z. Zhang, C. Wang, and L. Chen, “Continuous sign language recognition via Transformer networks,” Neurocomputing, vol. 453, pp. 568–577, 2021.
11. [11] A. Dosovitskiy et al., “Vision Transformers,” in Proc. ICLR, 2021.
12. [12] S. Faccio and M. Munaro, “Domain adaptation for real-time sign recognition,” IEEE Trans. Multimedia, vol. 25, pp. 3318–3330, 2023.
13. [13] X. Wei, Y. Zhou, and Q. Wang, “Benchmarking large-scale sign datasets,” IEEE Access, vol. 11, pp. 102134–102149, 2023.
14. [14] L. Li, J. Liu, and Y. Zhang, “Transformer-based sign recognition with spatio-temporal attention,” Pattern Recognit., vol. 144, 2023.
15. [15] Y. Zhao, Z. Li, and H. Wang, “Lightweight CNN for real-time gesture recognition,” IEEE Access, vol. 10, pp. 13424–13436, 2022.
16. [16] M. Abouelenien et al., “Multimodal gesture recognition,” Pattern Recognit., vol. 131, 2022.
17. [17] A. Moldovan, A. Popescu, and D. Maniu, “DL techniques for sign recognition: A review,” Expert Syst. Appl., vol. 193, 2022.
18. [18] N. Ding and Y. Wang, “ViT-based hand gesture recognition for edge,” Neural Netw., vol. 165, pp. 541–553, 2024.
19. [19] J. Kim, S. Lee, and H. Kim, “Real-time multilingual sign-to-speech,” ACM Trans. Accessible Comput., vol. 16, no. 1, 2024.
20. [20] Y. Xie, J. Lin, and H. Xu, “Robust gesture recognition in unconstrained environments,” IEEE Trans. Human-Machine Syst., vol. 54, no. 1, pp. 63–74, 2024.
21. [21] B. Shi, J. Chu, and X. Lin, “Cross-user sign recognition using attention networks,” IEEE Access, vol. 12, pp. 22134–22146, 2024.
22. [22] R. Ronchetti et al., “WLASL: A large-scale dataset for word-level ASL,” in Proc. IEEE CVPR Workshops, 2020.
23. [23] S. Escalera et al., “ChaLearn gesture dataset,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 10, pp. 2077–2090, 2016.
24. [24] M. Cooper et al., “Dataset challenges for sign recognition,” Signal Process.: Image Commun., vol. 94, 2021..

.