International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Real-Time Sign Language Translation System for Two-Way Communication

Purva Kanthe, Shrawani Khatpe, Mansi Pawar, Sakshi Shingate
Guide name: **Sakshi Paralkar**
Department of Artificial Intelligence and Machine Learning
Zeal Polytechnic, Pune, Maharashtra, India

ABSTRACT

Communication between people who use sign language and people who do not understand it can be difficult when a qualified interpreter is unavailable. This paper proposes a real-time, camera-based sign language translation system designed to support two-way interaction. In the sign-to-language direction, video frames are processed to detect hands and relevant movement, a recognition model classifies a supported sign or sign sequence, and the result is presented as text with optional speech synthesis. In the reverse direction, typed text or recognized speech is processed and mapped to a sign-language-specific visual representation, such as a verified sign clip or an articulated avatar. The design separates hand tracking, recognition, language processing, and output modules so that components can be tested and replaced independently. MediaPipe-style landmark extraction and machine-learning classifiers are considered for a constrained prototype, while continuous sign-language translation requires temporal modeling and linguistic context. The paper describes the architecture, methodology, evaluation plan, applications, limitations, and future development. No experimental performance is claimed because implementation details, dataset measurements, and test results were not supplied. The proposed system is therefore presented as a research and implementation framework rather than a validated product.

Keywords—Sign language recognition, real-time translation, computer vision, machine learning, gesture recognition, speech synthesis, two-way communication.

I. INTRODUCTION

Sign languages are natural languages expressed through coordinated hand shapes, movements, locations, body posture, and non-manual signals such as facial expression. They are not universal gesture codes: different communities use distinct sign languages, vocabularies, and grammatical conventions. When a conversation partner does not know the sign language used by a signer, everyday interactions in education, healthcare, public services, and workplaces may depend on writing, gestures, or access to an interpreter.

Computer vision and machine learning have created opportunities to recognize visual patterns from camera input. However, a practical communication aid must do more than classify a static hand pose. Many signs are dynamic, similarlooking signs may differ in movement or location, and meaning can depend on facial expression, body movement, sentence context, and the language community. Research distinguishes sign language recognition (identifying signs or glosses) from sign language translation (producing a natural-language sentence that preserves meaning) [1], [2].

This paper presents a modular design for a real-time two-way system. The first direction maps supported signs to readable text and optional synthesized speech. The second accepts text or speech and displays corresponding signs through a signlanguage-specific visual resource. The design emphasizes transparency: a limited-vocabulary prototype must not be described as unrestricted translation, and a word-to-clip lookup is not equivalent to generating grammatically correct sign language.

II. PROBLEM STATEMENT AND OBJECTIVES

A. Problem Statement

Communication is restricted when a signer and a non-signer do not share a language. Human interpreters may not be immediately available, written communication may be inconvenient, and lip-reading or improvised gestures can be ambiguous. Existing recognition prototypes are often constrained by vocabulary size, recording conditions, signer variation, and their reliance on isolated signs. Reverse systems can also fail when they map each spoken word independently without accounting for sign order, grammar, and non-manual information.

B. Objectives

The project objectives are: (1) capture sign activity using an ordinary camera; (2) detect and represent hand landmarks and, where available, body and facial cues; (3) classify signs from a defined sign-language dataset; (4) convert recognized signs into readable text and optional speech; (5) accept typed text or speech input in the reverse direction; (6) display appropriate sign videos or animations for a specified sign language; and (7) establish an evaluation protocol for recognition quality, latency, robustness, and usability. The first prototype should clearly define its target sign language and vocabulary before any performance claims are made.

III. RELATED WORK

Vision-based sign recognition has progressed from handcrafted image features and instrumented gloves to deep neural networks that learn spatial and temporal representations from video. Li et al. introduced WLASL, a large-scale word-level American Sign Language video dataset containing more than 2,000 words performed by over 100 signers, illustrating the importance of vocabulary coverage and signer diversity [3]. A dataset of this kind is useful for research but does not automatically provide a model that generalizes to every camera, signer, or sign language.

Zhang et al. presented MediaPipe Hands, a real-time pipeline combining palm detection with a hand-landmark model from RGB camera input [4]. Landmark coordinates can reduce the need to classify raw pixels directly and can support compact classifiers. Nevertheless, hand landmarks alone omit some linguistic information, including facial grammar, torso movement, and context. Landmark-based approaches should therefore be treated as a practical baseline for a limited task, not as a complete representation of sign language.

Camgoz et al. formulated sign language translation as a neural machine translation problem, learning a mapping from sign videos to spoken-language output rather than treating the task only as isolated gesture classification [1]. Stoll et al. explored text-to-sign production using neural machine translation and generative methods [5]. These studies show that translation and production are separate research problems, with the reverse direction requiring linguistic modeling and a suitable visual realization strategy.

Broader surveys report continuing challenges around benchmark availability, variation between datasets, multimodal cues, and evaluation [2], [6]. This project follows a staged approach: begin with a clearly bounded recognition vocabulary, then extend to temporal sequences and language-aware translation when appropriate data and resources are available.

IV. PROPOSED SYSTEM

The proposed system contains two linked but independently testable pipelines. The sign-to-text pipeline receives video frames, detects hands and optional pose/face landmarks, normalizes features, classifies a sign or sequence, applies temporal smoothing, and presents a text result. A text-to-speech engine may read the result aloud. The text/speech-to-sign pipeline accepts typed input or uses automatic speech recognition, performs language normalization and sentence-level mapping, then selects a validated sign sequence or generates a sign animation. The output is displayed visually, with text retained as a fallback.

The architecture should be configured for one named sign language, such as Indian Sign Language (ISL) or American Sign Language (ASL), rather than mixing signs from different languages. The target language is a project decision and must be reflected in the dataset, labels, translation rules, and sign media. The system should display an uncertainty or 'not recognized' state when confidence is low instead of silently emitting a potentially incorrect translation.

A.System Architecture

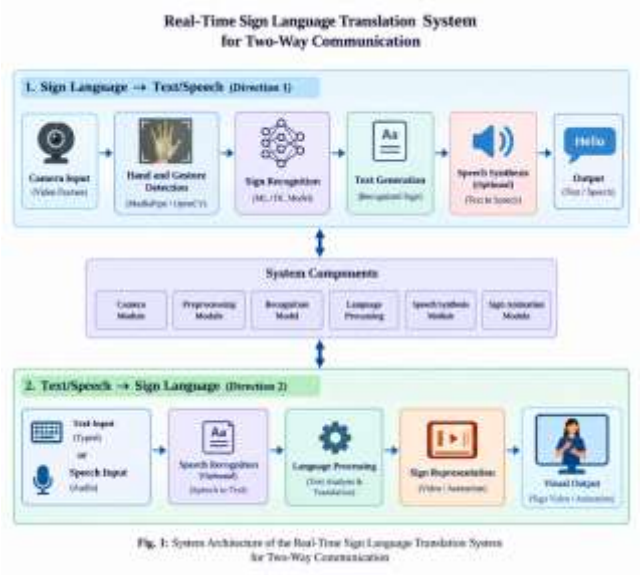


Fig. 1: System Architecture of the Real-Time Sign Language Translation System for Two-Way Communication

B. Main Components

Input module: captures camera frames or accepts typed/spoken language. Preprocessing: resizes frames, standardizes landmark coordinates, checks whether a hand is visible, and optionally filters noisy detections. Recognition module: uses a baseline classifier for static signs or a temporal model for moving signs. Language/output module: stabilizes predictions, builds text, and optionally invokes text-to-speech. Reverse translation module: processes sentences and maps them to a sign-language-specific sequence. Presentation module: shows confidence, recognized text, audio controls, sign media, and a way for the user to correct or repeat an input.

V. METHODOLOGY

A. Dataset Selection and Preparation B. Preprocessing and Feature Extraction

For a landmark-based baseline, video frames are passed to a hand detector and landmark estimator. Coordinates are translated relative to a stable reference point, scaled by a hand-size or bodyscale estimate, and checked for missing points. Temporal signs require a sequence of normalized landmarks across multiple frames; a single still frame may not preserve the movement that distinguishes a sign. If available and permitted by the task, pose and facial landmarks can be added as separate feature streams.

C. Model Training and Recognition

A small isolated-sign prototype can begin with a support vector machine, random forest, or multilayer perceptron trained on landmark features. A sequence model such as a temporal convolutional network, recurrent neural network, or transformer is more appropriate when movement and sign order matter. Model selection should be based on held-out validation data, class balance, inference cost, and error analysis—not on training accuracy alone. Confidence thresholds and a temporal voting or smoothing mechanism can reduce flickering predictions, although they cannot eliminate systematic model errors.

D. Text and Speech Generation

The recognized class is mapped to a label or phrase from the defined vocabulary. A sentence assembler may combine recognized units, but grammar-sensitive translation requires a trained sign-to-text model or carefully evaluated linguistic module. Optional speech output uses a text-to-speech engine after the text has been reviewed or confirmed by the user. The interface should allow cancellation and correction because recognition errors can materially change meaning.

E. Reverse Translation

Typed text can be processed directly; speech input first passes through an automatic speech recognition component. The text is normalized and interpreted in context before being mapped to a sequence in the target sign language. A basic prototype may retrieve verified clips for a restricted set of phrases, clearly labeling this as phrase-level playback. General-purpose text-tosign translation requires language-specific data, grammatical ordering, non-manual markers, coarticulation, and a natural visual realization. If no valid sign sequence exists, the system should request rephrasing or provide text rather than inventing a sign.

F. Proposed Technologies

Python is suitable for prototyping and model integration. OpenCV can manage camera capture and image operations; MediaPipe can provide hand or pose landmarks; scikit-learn can support classical baselines; and a deep-learning framework can implement temporal models. Speech recognition and text-tospeech components may be integrated through local or online services, subject to privacy, network, and licensing constraints. Sign clips should be sourced with permission and annotated by knowledgeable sign-language users. Avatar generation is a laterstage option, not an assumed implemented feature.

VI. ALGORITHMIC WORKFLOW

Algorithm 1: Sign-to-text inference. (1) Read a camera frame. (2) Detect the hand and estimate landmarks. (3) Normalize landmarks and append them to a bounded temporal buffer. (4) When the buffer satisfies the classifier's input requirements, infer a sign label and confidence. (5) Apply a confidence threshold and temporal stabilization. (6) Display a label or an uncertainty message. (7) On confirmation or a detected phrase boundary, append the label to the text output and optionally synthesize speech. (8) Clear or slide the buffer and continue.

Algorithm 2: Text/speech-to-sign output. (1) Accept typed text or capture speech. (2) If speech was supplied, transcribe it and allow correction. (3) Normalize the sentence and identify its intended meaning. (4) Generate or retrieve a target-language sign sequence using a sign-language-specific resource. (5) Validate that all required signs and transitions are available. (6) Play the corresponding clips or animation, with synchronized text where useful. (7) If the sentence cannot be mapped reliably, show a limitation message and retain the original text.

A. Technology Mapping

Table I lists candidate components. The word 'proposed' is intentional: the project implementation status was not provided, so no component is asserted to have been completed.

Component	Candidate technology	Role / status
Video input	OpenCV	Camera capture and frame handling; proposed
Landmark detection	MediaPipe Hands / Pose	Spatial features; proposed
Classifier	SVM, Random Forest, MLP	Baseline for bounded vocabulary; to be selected
Temporal model	LSTM / TCN / Transformer	Sequence recognition; extension requiring data
Speech input/output	ASR + TTS library/service	Optional modules; privacy and language support vary
Sign visual output	Licensed clips / avatar	Target-language sequence; requires validation

VII. EVALUATION PLAN AND DISCUSSION

No project dataset, trained model, test protocol, or measured results were supplied with the request. Accordingly, this paper does not report accuracy, frame rate, latency, or user-study findings as achieved results. The following evaluation plan should be completed after implementation.

Recognition should be evaluated on a signer-independent test set using top-1 accuracy, macro-averaged precision, recall, F1score, and a confusion matrix. For continuous sequences, word error rate or an appropriate sequence-level metric can supplement frame-level scores. End-to-end latency should be measured from camera frame acquisition to stable text output, reporting median and 95th-percentile delay on documented hardware. Real-time behavior should be measured rather than inferred from the use of a fast landmark detector.

Table II provides a results template to be filled with measured values. Leaving values blank is preferable to inventing evidence.

VIII. ADVANTAGES

1. Improved Communication: Enables communication between deaf or hard-of-hearing individuals and people who do not understand sign language.
2. Two-Way Communication: Supports communication in both directions by converting sign language gestures into text or speech and converting spoken or typed language into text or visual sign representations, depending on the system's features.

IX. APPLICATIONS

1. Education: Helps deaf and hard-of-hearing students communicate with teachers and classmates. It can support inclusive learning environments by translating signs into text or speech and spoken language into visual signs.
2. Healthcare: Enables patients and healthcare professionals to communicate more easily during consultations, medical examinations, and routine hospital visits.

X. LIMITATIONS

Recognition quality depends on training coverage and the camera view. Similar hand configurations, occlusion, motion blur, lighting changes, and signing styles can produce errors. Hand-only systems may miss facial expressions, mouth patterns, torso movement, and spatial references. A classifier trained on isolated signs may not understand continuous signing or sentence grammar. Sign languages differ across communities and should not be treated as interchangeable. The reverse direction is particularly challenging: word-by-word clip lookup may yield unnatural or incorrect sign sequences. Dataset licensing, privacy, consent, network availability, compute resources, and the quality of sign media also affect deployment.

XI. FUTURE SCOPE

Future work can expand the vocabulary and support continuous signing through temporally aware models. Multimodal architectures could combine hand, pose, face, and contextual information. Additional directions include building consent-based datasets for under-resourced sign languages, adapting models to new signers, on-device inference for privacy and offline use, and mobile deployment. Reverse translation may be improved through sign-language-specific neural translation and more natural avatar motion, but these advances require suitable parallel data and expert linguistic evaluation. Human correction, calibrated uncertainty, and accessible interface design should remain part of the system.

XII. CONCLUSION

This paper has described a proposed real-time sign language translation framework for two-way communication. Its sign-totext/speech pipeline combines camera input, landmark-based feature extraction, sign classification, text generation, and optional speech synthesis. Its reverse pipeline accepts text or speech and produces a target-language sign representation using

validated clips or a future animation module. The design separates recognition from translation and avoids claiming that isolated gesture classification or word lookup provides unrestricted natural sign language. Since no implementation measurements were supplied, empirical performance remains to be established through a signer-independent evaluation, latency testing, robustness analysis, and review by sign-language users. With language-specific data and careful validation, the system could become a useful assistive communication prototype.

ACKNOWLEDGMENT

The authors would like to thank [Project Guide Name], [Department/College Name], and the sign-language users or consultants who may contribute feedback and data to the project. This acknowledgment should be updated to reflect actual contributions and permissions; no contributor is implied to have participated unless confirmed by the authors.

REFERENCES

[1] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, “Neural sign language translation,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2018, pp. 7784–7793.

[2] “Sign language translation: A survey of approaches and techniques,” Electronics, vol. 12, no. 12, art. 2678, 2023, doi: 10.3390/electronics12122678.

[3] D. Li, C. Rodriguez, X. Yu, and H. Li, “Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison,” in Proc. IEEE/CVF Winter Conf. Applications of Computer Vision (WACV), 2020, pp. 1459–1469.

[4] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, “MediaPipe Hands: On-device real-time hand tracking,” arXiv:2006.10214, 2020.

[5] S. Stoll, N. C. Camgoz, S. Hadfield, and R. Bowden, “Text2Sign: Towards sign language production using neural machine translation and generative adversarial networks,” Int. J. Comput. Vis., vol. 128, pp. 891– 908, 2020, doi: 10.1007/s11263-019-01281-2.

[6] A. Wali, R. Shariq, S. Shoaib, S. Amir, and A. A. Farhan, “Recent progress in sign language recognition: A review,” Machine Vision and Applications, vol. 34, art. 127, 2023, doi: 10.1007/s00138-023-01479-y.