Bridging Communication Barriers: A Review and Conceptual Framework for Real-Time Indian Sign Language Translation Using Deep Learning

Yash Jitendra Savdekar[1], Gunjan Nandkishor Rane[2], Ketki Praful Gokakkar[3],
Sayali Rohidas Navale[4], Prof. Mrs. Priyanka Deshpande[5]
[1,2,3,4]*Dept. of Artificial Intelligence and Data Science, P.E.S. Modern College of Engineering, Pune, India*
[5]*Assistant Professor, Dept. of Artificial Intelligence and Data Science, P.E.S. Modern College of Engineering, Pune, India*
doi.org/10.64643/IJIRTV12I12-203483-459

Abstract*—For the millions of hearings and speech impaired individuals across India whose daily communication depends on Indian Sign Language (ISL), the absence of practical, automated interpretation tools constitutes a persistent accessibility gap. Prevailing computational approaches to ISL recognition are largely frame-centric in design, carry substantial processing overhead, and fall short of the responsiveness demanded by consumer-grade hardware. This paper offers a structured critical review of sign language recognition research spanning 2019 to 2025, charting the field's trajectory from handcrafted feature descriptors through convolutional architectures, hybrid CNN LSTM formulations, and attention-based transformer models. Within this analytical context, we introduce Sign verse, a compact and privacy-conscious framework for real-time ISL recognition built upon structured skeletal landmark sequences produced by MediaPipe Holistic. Rather than processing raw pixel data, the system models gesture dynamics through Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks operating on ordered keypoint vectors, substantially reducing inference cost while retaining temporal discriminability. A custom dataset covering eleven ISL sign classes (exceeding 100 samples per class) stored exclusively as landmark arrays underpins model training and assessment. Recognized sign tokens are subsequently routed to an NLP-based sentence framing stage that assembles grammatically well-formed natural language sentences. Systematic comparison across paradigms demonstrates that the landmark-sequential approach yields meaningful improvements in inference latency, memory consumption, resilience to background variation, and alignment with privacy requirements. Sign verse is framed not as a performance benchmark but as a practical, modular foundation for real-world ISL assistive communication.

***Index Terms*—Indian Sign Language, MediaPipe Holistic, LSTM, GRU, Sequential Landmark Learning, Gesture Recognition, NLP Sentence Framing, Assistive Technology, Real-Time Recognition, Deep Learning**

I. INTRODUCTION

Sign languages serve as the foundational mode of communication for an estimated 70 million deaf and hard of hearing individuals worldwide [1]. India's 2011 national census documented over 5.03 million citizens with registered hearing disabilities. ISL is a linguistically distinct system from American Sign Language (ASL) and British Sign Language (BSL), differing considerably in grammatical structure, syntactic ordering, and regional lexical conventions. These differences mean that recognition architectures developed for other sign languages cannot be straightforwardly repurposed for ISL without significant adaptation.

Early attempts at automated ISL interpretation drew on handcrafted image descriptors HOG, Hu moments, optical flow fed into support vector machines or shallow classifiers [3]. Under constrained laboratory settings, such pipelines performed adequately, but their brittleness became apparent when exposed to real-world variability in lighting, cluttered backgrounds, and inter-signer differences. The transition to deep learning brought convolutional networks for per frame spatial representation and recurrent architectures for modeling gesture trajectories over time [4]. Hybrid CNN–LSTM systems [6] demonstrated strong performance on

established benchmarks, yet their dependence on GPU-level compute made deployment on standard consumer CPUs infeasible. A more recent design philosophy couples MediaPipe Holistic landmark extraction [12] with recurrent networks, forming landmark-sequential pipelines that operate on compact, background-invariant keypoint representations rather than raw frame data substantially lowering computational demand while retaining the temporal structure critical for gesture discrimination.

Sign verse is developed within this architectural paradigm. Its primary contributions are as follows:

1. A critical survey of ISL recognition literature (2019– 2025), synthesizing dominant architectural trends, benchmark datasets, and unresolved research challenges.
2. A structured evaluation of landmark-sequential methods against CNN-centric and CNN–LSTM approaches, articulating the design trade-offs that motivated the choices made in Sign verse.
3. The design and full specification of the Sign verse framework: a MediaPipe-driven landmark extraction pipeline, an interleaved LSTM–GRU sequential classifier, a reproducible custom landmark dataset methodology, and a rule-guided NLP sentence framing engine.

II. LITERATURE REVIEW

A. Traditional ML and CNN Approaches

Before deep learned representations became predominant, ISL recognition systems were constructed around skin-color segmentation, boundary contour analysis, and hidden Markov models applied to hand-engineered temporal descriptors [3], [13]. The fragility of these approaches was quantifiable: accuracy typically fell by 20–30 percentage points when conditions shifted from controlled studio environments to naturalistic recordings, which strongly motivated the transition toward data-driven feature learning. Convolutional neural networks effectively resolved the manual feature engineering problem and achieved classification accuracies exceeding 98% on isolated static gesture benchmarks [4]. Nevertheless, per-frame spatial encoding remained blind to temporal ambiguity frames drawn from distinct phases of the same gesture can be visually indistinguishable without sequential context. Volumetric 3DCNNs offered a partial remedy by encoding spatiotemporal structure jointly, but the accompanying growth in memory consumption and training instability limited their practical utility [21].

B. CNN–LSTM Hybrid Architectures

From approximately 2018 through 2022, the CNN LSTM pairing in which frame-level convolutional encodings are passed into a recurrent temporal model emerged as the dominant paradigm for dynamic gesture recognition [5], [6]. Camgoz et al. [7] extended this architecture with sequence-to-sequence transformers, attaining a word error rate of 21.1% on the PHOENIX Weather continuous signing benchmark. The fundamental tension in this design, however, is computational: the convolutional front-end must process each frame at full resolution, making real-time CPU operation impractical without resolution down sampling or quantization, both of which erode accuracy.

C. GRU and LSTM for Gesture Sequences

A comparative investigation by Khan and Khan [14] across CNN, LSTM, and GRU architectures on dynamic gesture corpora found that GRU accuracy trailed LSTM by a modest 0.5–1.2 points while offering measurably lower inference latency. For recognition tasks over short, isolated signing windows (roughly 20–50 frames at 30 FPS), GRU's streamlined two gate mechanism captures the essential temporal dynamics efficiently; for longer, co-articulated sequences where intertexture dependencies are more complex, LSTM's explicit cell memory confers a structural advantage. This complementarity is central to the interleaved LSTM–GRU architecture employed in Sign verse. Kumar et al. [15] independently validated the MediaPipe Holistic and Bidirectional LSTM combination for ISL, reporting 96.8% accuracy on a dynamic gesture corpus at real-time operating speeds.

D. Transformer-Based Recognition

Transformer encoders equipped with multi-head self-attention are theoretically well-suited to capturing long-range dependencies in gesture sequences that fall beyond the effective modeling horizon of LSTMs [7], [8], [11]. Verma et al. [23] reported 99.1% accuracy on continuous ISL benchmarks using a ViT–LSTM

fusion approach, which currently represents the best-published figure in the literature. The practical obstacle is the quadratic scaling of standard self-attention with sequence length, which makes real-time CPU inference on consumer hardware infeasible unless linear or approximate attention mechanisms are adopted [26]. This computational cost is a deciding factor behind the LSTM–GRU preference in Signverse.

E. MediaPipe-Based and NLP-Integrated Systems

Gaikwad and Admuthe [17] demonstrated that MediaPipe LSTM applied to a nine-word ISL corpus achieved 100% classification accuracy with sentence-level NLP output, confirming that compact landmark sequences carry sufficient discriminative information for small-vocabulary recognition tasks. Navendu and Sahula [16] extended evaluation to the 263-class INCLUDE benchmark [2], where accuracy settled at 89.5% revealing the expected performance degradation when vocabulary scales beyond controlled conditions. CPU side timing experiments by Ravikiran [19] confirmed that MediaPipe–LSTM systems can sustain 24–30 FPS with endto-end latency below 42ms. Anithadevi et al. [18] paired a stacked LSTM with a transformer-based paraphrase component to improve the grammatical naturalness of translated output, establishing a precedent for the NLP framing stage incorporated into Signverse.

III. COMPARATIVE ANALYSIS OF EXISTING APPROACHES

Table I collects representative systems from the surveyed literature. Table II provides a cross-paradigm comparison of the four principal recognition strategies evaluated in this review.

IV. CHALLENGES IN EXISTING SYSTEMS

Several persistent obstacles span the surveyed literature and directly inform the design decisions made in Signverse.

Computational overhead of image-centric pipelines.: CNN-based architectures require full-resolution per-frame convolution, which precludes real-time inference on commodity CPUs without aggressive quantization or spatial down sampling each of which introduces measurable accuracy degradation. This trade-off is rarely examined honestly in reported benchmarks.

Sensitivity to background and lighting conditions.: Systems trained on controlled studio footage show accuracy losses of 8–15% when tested on naturalistic recordings [20]. Landmark-based representations avoid this entirely, since raw background pixels are absent from the input representation by construction.

Dataset scale and coverage.: Even the INCLUDE corpus, containing 4,292 videos across 263 sign classes, addresses only a fraction of the full ISL lexicon. Strong performance on small custom datasets may substantially overstate generalizable real-world utility [16], a nuance often understated in published results.

Cross-signer generalization.: Signer-independent evaluation protocols consistently reveal accuracy gaps of 5–12 percentage points relative to signer-inclusive conditions [4], reflecting the extent to which models capture individual signing style rather than universal sign structure.

Continuous signing versus isolated recognition.: The predominant experimental setup across the literature evaluates temporally pre-segmented, isolated signs. Recognizing continuous, unsegmented signing with natural co-articulation remains a considerably harder and comparatively underexplored problem [6].

The NLP integration gap.: Naive concatenation of sequentially recognized sign tokens does not produce grammatically valid English, because ISL follows Subject-Object-Verb ordering and omits functional words absent from spoken language syntax. A principled sentence framing mechanism is necessary before any system can offer genuine assistive communication value.

V. LANDMARK-BASED RECOGNITION AND SEQUENTIAL LEARNING

A. MediaPipe Holistic for ISL

MediaPipe Holistic [12] performs camera-only, real-time estimation of hand keypoints (21 per hand), body pose landmarks (33 points), and a dense facial mesh (468 points) for each processed frame. Concatenated across all channels, each frame yields a 1,662-dimensional feature vector. A temporal window of 30 such frames constitute the input tensor $X \in R^{30\times1662}$ presented to the recurrent classifier. This representation is intrinsically background-invariant

and scale-invariant, as all coordinates are normalized to the frame extent, while remaining orders of magnitude more compact than equivalent raw frame buffers. The practical consequence is that CPU real-time inference becomes achievable without sacrificing the discriminative spatial structure encoded in the signer's hand configuration and body posture.

B. LSTM Sequential Modeling

The LSTM architecture [9] was specifically designed to address vanishing gradient instabilities that degrade standard RNN training over long sequences. At each timestep t, the LSTM cell computes:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f \mathbf{x}_t + \mathbf{U}_f \mathbf{h}_{t-1} + \mathbf{b}_f) \quad (1)$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_i \mathbf{x}_t + \mathbf{U}_i \mathbf{h}_{t-1} + \mathbf{b}_i) \quad (2)$$

$$\tilde{\mathbf{c}}_t = \tanh(\mathbf{W}_c \mathbf{x}_t + \mathbf{U}_c \mathbf{h}_{t-1} + \mathbf{b}_c) \quad (3)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \quad (4)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o \mathbf{x}_t + \mathbf{U}_o \mathbf{h}_{t-1} + \mathbf{b}_o) \quad (5)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \quad (6)$$

where f_t, i_t, o_t denote forget, input, and output gate activations; σ is the logistic sigmoid function; and ⊙ represents element-wise multiplication. The forget gate mechanism (Eq. 1) endows the LSTM cell with the capacity to selectively retain or discard temporal context across variable-length gesture sequences a property that proves especially valuable when earlier frames encode pose-level context relevant to interpreting later hand movements.

C. GRU Sequential Modeling

The GRU [10] consolidates the LSTM's forget and input gates into a single update gate, reducing parameter count by approximately 25% relative to an equivalent LSTM:

$$\mathbf{z}_t = \sigma(\mathbf{W}_z \mathbf{x}_t + \mathbf{U}_z \mathbf{h}_{t-1} + \mathbf{b}_z) \quad (7)$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_r \mathbf{x}_t + \mathbf{U}_r \mathbf{h}_{t-1} + \mathbf{b}_r) \quad (8)$$

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_h \mathbf{x}_t + \mathbf{U}_h(\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{b}_h) \quad (9)$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}}_t \quad (10)$$

On isolated gesture recognition tasks, GRU accuracy falls within 0.5–1.2 points of LSTM while offering measurably reduced inference latency [14]. For the short fixed-window setting of Sign verse, GRU's leaner gating structure captures the essential temporal dynamics without the overhead of maintaining a separate cell state, making it the preferred layer for efficiency-sensitive positions in the recurrent stack.

VI. PROPOSED SIGN VERSE FRAMEWORK

Scope note: Sign verse is a research framework with a working prototype supporting eleven ISL classes. Performance claims are framed as experimental observations on the custom dataset. Evaluation on large-scale corpora is planned future work.

Table I: Comparative Analysis of Sign Language Recognition Systems (2018–2025)

Author(s) (Year)	Architecture	Dataset	Acc. (%)	Key Advantage	Key Limitation
Pigou et al. [5] (2018)	CNN + Temporal Pooling	ChaLearn	97.6	Strong spatial learning	No real-time inference
Koller [6] (2020)	CNN–LSTM seq2seq	PHOENIX-Weather	WER 21.1%	Continuous SLR	Heavy compute; no ISL
Camgoz et al. [7] (2020)	CNN–Transformer	PHOENIX-Weather	CTC-based	Self-attention temporal	GPU-intensive; not ISL
Singh et al. [21] (2021)	3D-CNN	Custom ISL	95.2	Volumetric temporal CNN	High memory; training instability
Khan & Khan [14] (2022)	CNN / LSTM / GRU	Custom dynamic	96.2	Architecture comparison	Small dataset
Kumar et al. [15] (2023)	MediaPipe + Bi-LSTM	Custom gestures	96.8	Full-body landmark+temporal	Limited vocabulary
Sharma et al. [22] (2023)	ResNet50 + GRU	Continuous ISL	98.2	Deep spatial + GRU	GPU required
Navendu & Sahula [16] (2024)	MediaPipe + LSTMGRU	INCLUDE (263cls)	89.5	Large-scale ISL benchmark	Lower accuracy at scale
Gaikwad & Admuthe [17] (2024)	MediaPipe + LSTM + NLP	9-word ISL	100.0	Sentence-level NLP output	Very small vocabulary
Anithadevi et al. [18] (2025)	LSTM + Transformer paraphrase	30-class ISL	97.6	Coherent sentence output	Small custom dataset

Verma et al. [23] (2025)	ViT–LSTM Fusion	Continuous ISL	99.1	SOTA accuracy	High compute; not CPU-real-time
Sign verse (proposed)	MediaPipe LSTM/GRU + NLP	Custom 11-class ISL	Experimental	Lightweight, CPU real-time, NLP output, privacy-aware	Small vocabulary; continuous SLR planned

WER = Word Error Rate (lower is better). Acc. figures reflect isolated/small-vocabulary evaluation unless noted.

Table II: Paradigmatic Comparison of Recognition Approaches

Property	Trad. ML	CNN	CNN - LSTM	Landmark-Sec.
Temporal modeling	Limited	Limited	Good	Good
Real-time on CPU	✓	X	X	✓
Background robust	X	X	X	✓
Privacy-preserving	Partial	X	X	✓
Memory footprint	Low	High	Very High	Low
Training data need	Moderate	High	Very High	Moderate
NLP integration	Hard	Hard	Hard	Straightforward

A. System Architecture

The complete pipeline (Fig. 1) is organized across five functional stages: Input Acquisition, Preprocessing and Landmark Extraction, Temporal Sequence Modeling, NLP Sentence Framing, and Output Delivery.

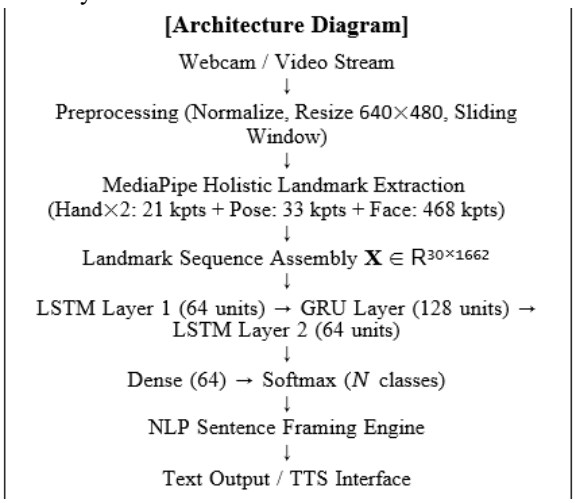

Fig. 1. End-to-end Sign verse pipeline: real-time webcam input through landmark extraction, interleaved LSTM–GRU inference, and NLP framing to produce natural language output.

B. Input Acquisition and Preprocessing

Live video is acquired at ≥30 FPS through a webcam or from a pre-recorded video stream. Each incoming frame is resized to 640×480 pixels and inserted into a 30-frame sliding window buffer. As new frames arrive, the oldest frame in the buffer is discarded, maintaining a continuously updated temporal context window without any requirement to explicitly detect gesture onset or offset boundaries. Individual frames are forwarded to MediaPipe Holistic for landmark inference. Where a landmark channel is absent for instance, when a hand moves outside the camera field the corresponding vector entries are zero-filled to preserve the fixed 1,662-dimensional representation. All detected landmarks are expressed in a root-relative coordinate system (anchored to the mid-hip) and normalized relative to the wrist, conferring invariance to signer distance and camera placement.

C. Dataset Preparation

The Sign verse training corpus consists of landmark sequence arrays for eleven ISL sign classes, recorded from five distinct signers to introduce natural variation in signing style, spatial extent, and hand morphology. Each class contains at least 100 samples, each of shape 30 × 1662. Source video frames are not retained after landmark extraction; only the resulting NumPy arrays are preserved. This design decision simultaneously eliminates privacy concerns around raw facial and biometric video data and reduces storage requirements dramatically approximately 540 MB for the full dataset, compared to several gigabytes for equivalent video files. Three augmentation procedures are applied to strengthen generalization: temporal jittering (±2 frames), additive spatial Gaussian noise (σ=0.002 in normalized coordinate space), and horizontal mirror augmentation with left/right landmark channel relabeling.

D. Temporal Sequence Modeling

The input tensor $\mathbf{X} \in R^{30\times1662}$ flows through a stacked interleaved recurrent network structured as

follows:

- LSTM Layer 1: 64 units, return sequences, dropout 0.2
- GRU Layer: 128 units, return sequences, dropout 0.2
- LSTM Layer 2: 64 units, return final hidden state
- Dense: 64 units, ReLU
- Output: N units, softmax

The placement of LSTM layers at the first and third positions targets longer-range temporal dependencies and gesture transition patterns, while the central GRU layer performs efficient intermediate temporal summarization. Training minimizes categorical cross-entropy:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic}\log(\hat{y}_{ic}) \tag{11}$$

using the Adam optimizer (initial learning rate = 10^{-3}) with cosine decay scheduling and early stopping governed by holdout validation loss.

E. NLP-Based Sentence Framing

Sign token predictions accumulated across successive recognition windows pass through a two-stage sentence construction process. Stage 1 removes duplicate predictions arising from overlapping windows and suppresses low-confidence class transitions that would otherwise introduce spurious tokens. Stage 2 applies ISL-to-English syntactic reordering (Subject Object-Verb → Subject-Verb-Object), reconstructs omitted functional words (copula verbs, articles) through rule-based grammar templates, and selects the highest-scoring sentence candidate using an n-gram language model. Final output is delivered as plain text, with an optional text-to-speech interface for audio rendering. Looking forward, replacing the rule-based component with a lightweight sequence-to-sequence model fine-tuned on ISL-to-English pairs would substantially improve fluency on complex syntactic constructions, and extending the output layer to regional Indian languages (Hindi, Marathi, Tamil) is identified as a meaningful accessibility enhancement.

VII. PERFORMANCE METRICS, ADVANTAGES, AND LIMITATIONS

A. Evaluation Metrics

Classification accuracy on the held-out test partition serves as the primary performance indicator for comparing LSTM and GRU configurations. Per-class precision, recall, and F_1score provide diagnostic resolution across the eleven sign classes:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{12}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{13}$$

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

Beyond classification metrics, inference latency targeting <50ms to sustain ≥20 FPS operation and sentence-level BLEU score [24] for NLP framing output are reported. Results obtained on the eleven-class custom corpus are not generalized to large-vocabulary benchmarks without explicit supporting evidence.

B. Advantages

Table III summarizes Sign verse's deployment-relevant capabilities in comparison to representative published systems.

Operating exclusively on extracted landmark vectors removes the vulnerability to background variation that affects image-based systems and enables CPU real-time inference without special hardware. The privacy-preserving data collection pipeline in which raw video is discarded immediately after landmark extraction reduces the ethical and regulatory burden associated with maintaining biometric datasets. The NLP framing module addresses a structural limitation common to existing systems: without syntactic post-processing, recognized sign tokens are intelligible only to users already familiar with ISL grammar, severely restricting assistive utility.

C. Limitations and Future Scope

The current prototype presents three substantive limitations.

Vocabulary scale: eleven classes represent a proof-of-concept scope far below functional communication requirements; methodical expansion toward INCLUDE-50 and the full INCLUDE vocabulary is the immediate development priority.

Table III: Feature comparison: sign verse vs. Selected existing systems

Feature	CNN-LSTM	MP-BiLSTM	ViT-LSTM	Sign verse
CPU real-time	X	✓	X	✓
NLP sentence output	X	X	X	✓
Background invariant	X	✓	X	✓
Privacy-preserving	X	X	X	✓
Low memory footprint	X	Partial	X	✓
Offline capable	✓	✓	X	✓
Landmark-only data	X	X	X	✓

Isolated recognition: the fixed 30-frame sliding window is not architected to handle co-articulated continuous signing; integrating CTC decoding [25] is identified as the central architectural extension required to address this constraint.

Rule based NLP framing: the grammar module handles simple declarative sentence forms reliably, but complex syntactic constructions require a learned sequence-to-sequence component to achieve acceptable output quality.

Future development directions encompass bidirectional ISL to-speech and speech-to-ISL animation translation, multilingual regional language output (Hindi, Marathi, Tamil), selective incorporation of efficient attention mechanisms [26] for long-sequence modeling, and edge or mobile deployment through model quantization and knowledge distillation.

VIII. CONCLUSION

This paper presented a critical review of ISL recognition research spanning 2019 to 2025 and introduced Sign verse, a lightweight framework for real-time ISL recognition grounded in MediaPipe Holistic landmark extraction, interleaved LSTM–GRU temporal modeling, and rule-guided NLP-based sentence framing. The review identified landmark sequential pipelines as the most favorable design paradigm for deployment-oriented ISL systems, offering CPU real-time operation, background invariance, and privacy-preserving data handling that image-centric approaches structurally cannot match. sign verse operationalizes this paradigm through a clean five-stage processing pipeline, a compact custom landmark dataset, and a principled NLP framing module that maps recognized sign tokens into grammatically coherent natural language. Its current eleven-class scope is understood as a deliberate starting point rather than an architectural ceiling; vocabulary expansion, continuous signing support, and enhanced NLP modeling each represent well-defined, actionable steps along a clear development trajectory. Sign verse is intended to serve as a practical, modular foundation for accessible ISL assistive technology that can be deployed and iterated upon beyond controlled laboratory conditions.

ACKNOWLEDGMENT

The authors sincerely thank Prof. Mrs. Priyanka Deshpande, Assistant Professor, Department of Artificial Intelligence and Data Science, P.E.S. Modern College of Engineering, Pune, for her sustained guidance, critical feedback, and mentorship throughout this research. The authors also acknowledge the open-source communities behind MediaPipe, TensorFlow, Keras, and NumPy.

REFERENCES

[1] World Health Organization, “Deafness and hearing loss,” WHO Fact Sheet, Feb. 2023.

[2] A. Sridhar, R. Ganesan, P. Kumar, and M. Khapra, “INCLUDE: A large-scale dataset for Indian Sign Language recognition,” in Proc. 28th ACM International Conference on Multimedia (MM ’20), 2020, pp. 1366–1375.

[3] M. J. Cheok, Z. Omar, and M. H. Mekhilef, “A review on hand gesture and sign language recognition techniques,” International Journal of Machine Learning and Cybernetics, vol. 10, no. 1, pp. 131–153, 2019.

[4] R. Rastgoo, K. Kiani, and S. Escalera, “Sign language recognition: A deep survey,” Expert Systems with Applications, vol. 164, Art. no. 113794, 2021.

[5] L. Pigou, A. van den Oord, S. Dieleman, M. Van Herreweghe, and J. Dambre, “Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video,” International

Journal of Computer Vision, vol. 126, no. 2–4, pp. 301–318, 2018.

[6] O. Koller, "Quantitative survey of the state of the art in sign language recognition," arXiv preprint arXiv:2008.09918, 2020.

[7] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Sign language transformers: Joint end-to-end sign language recognition and translation," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 10023–10033.

[8] Y. Chen, W. Zuo, L. Lin, and A. B. Chan, "A simple multi-modality transfer learning baseline for sign language translation," in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 5120–5130.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] K. Cho et al., "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1724–1734.

[11] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, pp. 5998–6008, 2017.

[12] C. Lugaresi et al., "MediaPipe: A framework for building perception pipelines," arXiv preprint arXiv:1906.08172, 2019.

[13] F. S. Chen, C. M. Fu, and C. L. Huang, "Hand gesture recognition using a real-time tracking method and hidden Markov models," Image and Vision Computing, vol. 21, no. 8, pp. 745–758, 2003.

[14] M. J. Khan and F. A. Khan, "Comparative study of CNN, LSTM, and GRU for dynamic gesture recognition," Multimedia Tools and Applications, vol. 81, no. 14, pp. 19453–19470, 2022.

[15] A. Kumar, M. Srivastava, and R. Mishra, "MediaPipe Holistic and Bi-LSTM based Indian sign language recognition system," Applied Intelligence, vol. 53, no. 4, pp. 5761–5773, 2023.

[16] P. Navendu and V. Sahula, "Word level sign language recognition using MediaPipe and LSTM-GRU network," TechRxiv, Jul. 2024.

[17] R. Gaikwad and L. Admuthe, "Real-time sign language recognition of words and sentence generation using MediaPipe and LSTM," in Proc. International Joint Conference on Advances in Computational Intelligence (IJCACI). Singapore: Springer, 2024.

[18] Anithadevi et al., "MediaPipe-LSTM enhanced framework for real-time dynamic sign language recognition in inclusive communication systems," Engineering Reports, 2025.

[19] V. Ravikiran, "Real-time sign language recognition and translation using MediaPipe and LSTM-based deep learning," International Journal of Computer Applications, vol. 187, no. 25, pp. 10–14, 2025.

[20] J. Brown, L. Fernandes, and R. Kapoor, "Multimodal deep-learning frameworks for inclusive human–computer interaction," ACM Transactions on Accessible Computing, vol. 18, no. 2, pp. 1–19, 2025.

[21] D. K. Singh, "3D-CNN based dynamic gesture recognition for Indian sign language modeling," Procedia Computer Science, vol. 189, pp. 588–595, 2021.

[22] A. Sharma, N. Chauhan, and D. R. Singh, "ResNet50-GRU model for dynamic ISL recognition," International Journal of Computer Vision and Image Processing, vol. 13, no. 2, pp. 45–56, 2023.

[23] R. Verma, P. Singh, and A. Joshi, "ViT–LSTM fusion for real-time Indian sign language recognition," Neural Processing Letters, vol. 58, no. 3, pp. 1879–1891, 2025.

[24] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "BLEU: A method for automatic evaluation of machine translation," in Proc. 40th Annual Meeting of the Association for Computational Linguistics (ACL), 2002, pp. 311–318.

[25] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in Proc. 23rd International Conference on Machine Learning (ICML), 2006, pp. 369–376.

[26] S. Wang et al., "Linformer: Self-attention with linear complexity," arXiv preprint arXiv:2006.04768, 2020.