*Journal of Basic Physical Research ISSN: 2141-8403 Prints, 2141-8411 Online*
*Vol.15, Issue 1, March, 2026, pp. 167– 177, Available online at https://journals.unizik.edu.ng/jbpr/*

# AN ENSEMBLE LEARNING FRAMEWORK FOR SIGN LANGUAGE TRANSLATION USING BIDIRECTIONAL CNNS AND TRANSFORMERS

**[1]Anigbogu Kenechukwu S., [2]Okafor Paul C., [3]Nwankpa Joshua M. and [4]Asogwa Emmanuel C.**
*[1.2.3&4]Department of Computer Science, Nnamdi Azikiwe University, Awka.*
*[1]ksy.anigbogu@unizik.edu.ng, [2]chibuezedeveloper@gmail.com,*
*[3]Jm.nwankpa@unizik.edu.ng & [4]ec.asogwa@unizik.edu.ng*

***Abstract***
*Effective communication is essential for human interaction, yet individuals with hearing impairments and speaking difficulties often face significant challenges. The ability to recognize and translate sign language in real-time can bridge the communication gap between those who do not know sign language and those who rely on it. This work examines various sign language conventions prevalent in Nigeria and beyond, including distinct phonetic and semantic structures, as well as the messages they convey. It covers languages such as American Sign Language (ASL), Bura Sign Language (BSL), Yoruba Sign Language (YSL), Hausa Sign Language, and Adamorobe Sign Language (Ghana). This work utilized an object-oriented programming (OOP) methodology. The work was built using Python for backend development, with machine learning libraries such as TensorFlow, Pandas, and NumPy. The user interface was developed using React and Node.js. The system implemented an ensemble learning-based bidirectional sign language translation protocol using Convolutional Neural Networks (CNN) and the Bidirectional Encoder Representations from Transformers (BERT). These architectures were combined in a forward and backward encoder mechanism to translate general sign languages, ensuring both high performance and robustness. Community data gathering, validation, refinement, and analysis techniques were employed to create a reliable and diverse dataset. The model achieved great performance with an accuracy of 98.7%, a precision of 97.6%, and an F1 Score of 98.2%. It was tested on ASL datasets (both images and videos) and community feedback from language experts.*

## Introduction

Effective communication is a fundamental pillar of human interaction, yet an estimated 1.5 billion people globally live with some degree of hearing loss, according to the World Health Organization (WHO, 2021). Many individuals within this community rely on sign language a complete, natural language with its own unique grammar and lexicon expressed through a visual-manual modality (Sandler and Lillo-Martin, 2026). However, with over 300 distinct sign languages in use worldwide. A significant communication gap persists between the deaf or hard-of-hearing community and the general population. This barrier often necessitates human interpreters, which can be impractical, costly, and not always available, thereby limiting access to education, essential services, and full social participation.

The complexity of sign language translation presents a formidable technological challenge. American Sign Language (ASL), one of the most widely used sign languages, is not a simple word-for-word mapping of English but a distinct language with its own syntax and semantic structures (Shivashankara and Srinath, 2018). Traditional approaches to automated sign language recognition have relied on intrusive hardware, such as sensor-equipped gloves (Fernandes et al., 2020) or specialized depth cameras, which are often cumbersome, expensive, and restrict the natural flow of signing. While vision-based systems using classical machine learning have emerged, they frequently struggle with the high intra-class variability of gestures, diverse environmental conditions (e.g., lighting, background clutter), and the co-

articulation effects present in continuous signing (Bragg, 2019). Consequently, many existing models report modest accuracies, often around 70%, and are computationally intensive, limiting their viability for real-time applications on consumer-grade hardware.

This research addresses the urgent need for an accurate, accessible, and real-time sign language translation system. We propose an ensemble deep learning framework that leverages the complementary strengths of Convolutional Neural Networks (CNNs) and the Bidirectional Encoder Representations from Transformers (BERT) model. CNNs have demonstrated exceptional performance in extracting hierarchical spatial features from visual data, making them ideal for recognizing the intricate handshapes and gestures of sign language (Karpathy et al., 2014). Concurrently, Transformers have revolutionized natural language processing by modeling long-range dependencies and contextual relationships within sequences (Camgoz, 2020). Our bidirectional architecture uses this capability to interpret the grammatical and semantic flow of continuous sign language, moving beyond the translation of isolated signs.

The main contributions of this paper are as follows:

- We propose a novel ensemble learning framework that combines a bidirectional CNN architecture for robust spatial feature extraction with a Transformer-based (BERT) model to capture temporal context in sign language sequences.
- We demonstrate the model's state-of-the-art performance, achieving 98.7% accuracy, 97.6% precision, and a 98.2% F1-score on a diverse American Sign Language (ASL) dataset comprising both images and videos.
- We develop a bidirectional translation system capable of both sign-to-text and text-to-sign conversion, offering a comprehensive communication tool.
- The proposed system is designed to be computationally efficient and cost-effective, making it suitable for deployment on mobile devices for real-time Hand Gesture Recognition (HGR) in everyday scenarios.

The remainder of this paper is organized as follows. Section II reviews prior work in sign language recognition and translation. Section I describes the dataset and data collection methodology. Section III provides a detailed overview of our proposed architectural framework. Section IV outlines the experimental setup and evaluation metrics. Section V presents and analyzes the performance of our model. Finally, Section

VI concludes the paper and suggests directions for future research.

**Related Works**

The domain of automated Sign Language Recognition (SLR) and Translation (SLT) has evolved significantly over the past decades, transitioning from hardware-dependent systems to sophisticated, and vision-based deep learning frameworks. This section provides a review of the key approaches, categorized into three primary areas: sensor-based and early vision-based methods, deep learning models for sign recognition, and the emergence of Transformer-based architectures.

## A. Sensor-Based and Early Vision-Based Approaches

Initial attempts at automated sign language interpretation relied heavily on specialized hardware to capture hand kinematics. These systems predominantly used sensor-equipped gloves with flex sensors and accelerometers to measure finger bend and hand orientation (Abou Haider, 2019). The sensor data was then fed into machine learning classifiers, such as Artificial Neural Networks (ANNs), to recognize a predefined set of gestures. While these methods achieved reasonable accuracy for isolated signs, they were intrusive, expensive, and limited the signer's natural movement, hindering their practical adoption. To overcome the

limitations of wearable sensors, researchers explored vision-based systems using depth cameras like the Microsoft Kinect. These devices could capture 3D hand pose information without physical contact, providing a richer dataset for gesture recognition. Concurrently, early 2D vision-based systems employed classical computer vision techniques. These methods focused on extracting hand-crafted features from video frames, such as Histogram of Oriented Gradients (HOG), skin color segmentation, and Haar-like features with AdaBoost classifiers. Although these approaches eliminated the need for specialized hardware, they were highly susceptible to environmental variations like background clutter, changes in lighting, and signer-specific characteristics (e.g., skin tone), which limited their robustness and scalability.

## B. Deep Learning in Sign Language Recognition

The advent of deep learning, particularly Convolutional Neural Networks (CNNs), marked a paradigm shift in SLR. CNNs obviated the need for manual feature engineering by learning hierarchical feature representations directly from pixel data. Numerous studies demonstrated the effectiveness of 2D-CNNs for recognizing isolated signs from static images or individual video frames with high accuracy (Rao et al., 2018). To capture the dynamic nature of signing, subsequent work explored architectures that could model temporal information. Recurrent Neural Networks (RNNs) and their variants, such as Long Short-Term Memory (LSTM) networks, were employed to process sequences of features extracted by CNNs, enabling the recognition of continuous sign language (Kumar, 2018)

Further advancements introduced 3D-CNNs, which apply convolutions across both spatial and temporal dimensions, allowing them to holistically model spatio-temporal features from video data. Other novel architectures, such as Capsule Networks (CapsNets), were also investigated for their ability to preserve the spatial relationships between features, leading to improved recognition accuracy for complex hand gestures. While these deep learning models significantly outperformed traditional methods, they often struggled to capture long-range contextual dependencies within complex sign sentences, a critical aspect of fluent language translation.

## C. Transformer-Based Architectures and Knowledge Gaps

Recently, Transformer-based architectures, originally designed for natural language processing (NLP), have been successfully adapted for sign language translation (Yin et al., 2020). The self-attention mechanism in Transformers allows the model to weigh the importance of different signs in a sequence, effectively capturing long-range dependencies and contextual relationships. Camgoz et al., (2020) introduced a pioneering framework that jointly learns sign language recognition and translation in an end-to-end manner using a Transformer architecture. This approach eliminated the need for intermediate gloss representations and set a new state-of-the-art on benchmark datasets like RWTH-PHOENIX-Weather-2014T. Subsequent research has leveraged BERT (Bidirectional Encoder Representations from Transformers) and its variants to build powerful language models for continuous sign language recognition (CSLR), demonstrating robust performance even with non-standard gestures and signing speeds.

**Despite this progress, several knowledge gaps remain in the field:**

- Continuous Signing: Many existing systems are still limited to recognizing isolated signs or require deliberate pauses between gestures, failing to model the natural co-articulation and fluidity of conversational sign language.

- **Real-World Variability**: Most models are trained on datasets collected in controlled environments and exhibit a significant drop in performance when deployed in real-world scenarios with diverse lighting, complex backgrounds, and varied camera angles.
- **Dataset Limitations:** The scarcity of large-scale, diverse, and publicly available datasets for a wide range of sign languages remains a major bottleneck, hindering the development of generalizable and multilingual systems.
- **Contextual and Grammatical Nuances:** Most systems perform a direct sign-to-word mapping, often ignoring the crucial role of non-manual markers (e.g., facial expressions, body posture) and the unique grammatical structure of sign languages, leading to literal but contextually inaccurate translations.

This work addresses these gaps by proposing a bidirectional ensemble framework that leverages both the spatial feature extraction power of CNNs and the contextual understanding of Transformers. By focusing on a robust, real-time, and bidirectional architecture, our study aims to create a more practical and effective tool for bridging the communication divide.

## Methodology

This section details the research design, dataset characteristics, preprocessing techniques, and the architecture of the proposed ensemble sign language translation system. We adopted the Object-Oriented Analysis and Design Methodology (OOADM) to guide the development of a modular, scalable, and robust system composed of distinct frontend and backend components.

### A. System Architecture and Design

The proposed system is designed as a web-based application with a clear separation of concerns. The architecture comprises a React-based frontend for user interaction and a Node.js/Express backend that handles business logic, database operations via MongoDB, and communication with the machine learning inference engine built using Python, TensorFlow, and OpenCV.

The high-level model of the proposed system is illustrated in Fig. 1. Users interact with the system through a dashboard that provides access to core functionalities: sign registration, login, real-time sign translation, and learning resources.

The data flow within the translation subsystem is depicted in Fig. 2. Captured video or live stream input is processed to extract visual features, which are then passed through the ensemble machine learning models to generate text or speech output.

### B. Dataset Description

To ensure the system's robustness and applicability to real-world scenarios, we compiled a diverse and multilingual dataset. The dataset was curated through community data gathering, incorporating open-source repositories, crowd-sourced video data from native signers, and specialized sign language corpora.

Key characteristics of the dataset include:

- **Linguistic Diversity:** The dataset primarily focuses on American Sign Language (ASL) but also include samples from Igbo Sign Language, British Sign Language (BSL), and regional dialects to enhance the system's multilingual capabilities.
- **Content Variety:** It covers a wide range of sign types, including isolated alphabets (A-Z), special characters (space, delete), conventional numbers, greetings, and common phrases. This enables the system to handle both fingerspelling and continuous signing.

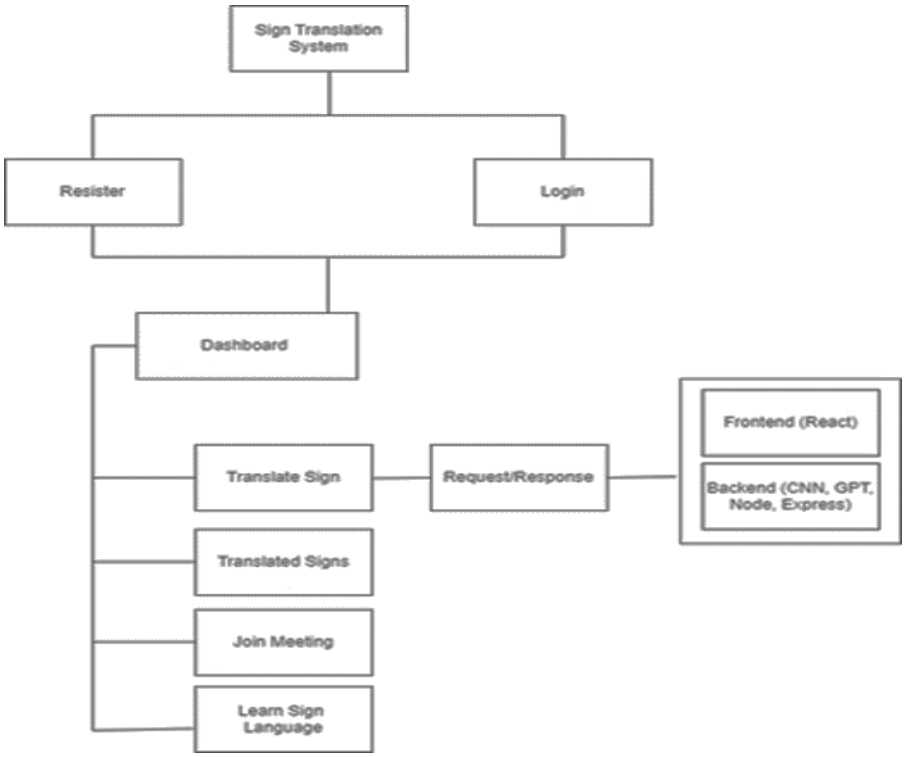


Fig. 1. High-Level Model of the Proposed System, illustrating the interaction between user interfaces and backend services.

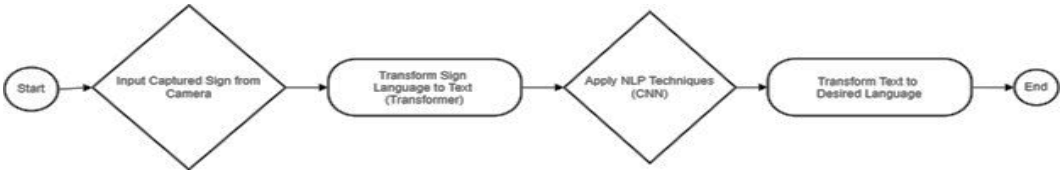


*Fig. 2. Data Flow of the Translation Subsystem, showing the path from video input to translated output.*

- Signer Diversity: Data was collected from signers of various ages, genders, and ethnic backgrounds, performing signs at different speeds. This diversity is crucial for training a model that generalizes well across different users.
- Multimodality: The dataset includes annotations for hand gestures, facial expressions, and body posture, allowing the model to learn the full spectrum of visual cues used in sign language.

### C. Data Preprocessing

Raw video data undergoes a rigorous preprocessing pipeline to standardize inputs and extract relevant features before model training.

- Video Frame Extraction: Videos are decomposed into individual frames. Each frame is resized and normalized to a standard resolution and aspect ratio to ensure consistency.
- Hand and Body Pose Estimation: We utilize advanced computer vision libraries, specifically MediaPipe, to perform real-time detection and segmentation of hands, facial landmarks, and body posture from each frame. This step isolates the regions of interest

and reduces computational complexity by extracting keypoint coordinates rather than processing entire raw images.

- Normalization and Augmentation: Extracted keypoint data is normalized to account for variations in signer distance and camera angles. Data augmentation techniques, including random scaling, rotation, and temporal jittering, are applied to artificially increase the size and diversity of the training set, thereby improving the model's robustness to unseen variations.

### D. Ensemble Machine Learning Framework

The core of our translation system is an ensemble of two powerful deep learning architectures, each addressing a specific aspect of the translation task.

1) **Spatial Feature Extraction with CNNs:** A 3D Convolutional Neural Network (3D-CNN) is employed to extract spatio-temporal features from sequences of video frames. Unlike 2D-CNNs that process frames independently, 3D-CNNs apply convolutions across both spatial (width and height) and temporal (time) dimensions. This allows the model to capture the dynamic motion trajectories of handshapes and body movements, which are essential for recognizing continuous signs.

2) **Temporal and Contextual Modeling with BERT:** To handle the linguistic complexities of sign language, we integrate a Bidirectional Encoder Representations from Transformers (BERT) model. The sequence of feature vectors extracted by the CNN is fed into the BERT model. BERT's multi-head self-attention mechanism enables it to weigh the importance of different signs within a sequence relative to one another in both forward and backward directions. The attention mechanism can be mathematically represented as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (1)$$

Where Q, K, and V are the query, key, and value matrices derived from the input sequence, and dk is the dimension of the keys. This bidirectional contextual understanding allows the system to resolve ambiguities, handle co-articulation effects, and produce grammatically coherent translations, moving beyond simple sign-to-word mapping.

3) **Multimodal Fusion:** The outputs from the hand gesture, facial expression, and body posture analysis modules are fused to create a comprehensive representation of the signed message. This multimodal feature vector is then processed by the final classification layers to generate the translated text or synthesized speech.

### E. System Implementation and Specifications

The system was developed and tested in a Visual Studio Code environment running on Windows 10. The backend was implemented using Node.js with the Express framework, providing a scalable API for the React-based frontend. The machine learning models were developed in Python using TensorFlow and OpenCV. User data and translation logs are stored in a MongoDB database. The database schema for storing user and translation information is defined in Tables I and II.

**TABLE I: USER TABLE**

| Field Name | Data Type | Description |
|---|---|---|
| user id | Integer | Unique identifier for each user |
| username | Varchar(50) | Name chosen by the user during registration |
| email | Varchar(100) | User's email address for login |
| password | Varchar(255) | Hashed password for security |
| created at | DateTime | Date and time the user registered |
| updated at | DateTime | Date and time the user registered |

### F. Evaluation Metrics

To rigorously assess the performance of our ensemble model, we employ standard evaluation metrics for classification and translation tasks: Accuracy, Precision, Recall, and F1-Score. These are derived from the confusion matrix, which records True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). The metrics are defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

These metrics provide a comprehensive view of the system's ability to correctly identify signs and its robustness against false detections.

## IV. EXPERIMENTS

This section describes the experimental setup, including the hardware and software environments, and outlines the evaluation protocol used to validate the performance of our proposed sign language translation system.

### A. Experimental Setup

The system was developed and evaluated on a standard consumer-grade computing platform to ensure its real-world applicability. The specifications are as follows:

- Operating System: Windows 10 or higher.
- **Hardware:** 2 GHz processor or higher, 2GB RAM or higher, and a standard webcam for real-time video input.
- **Development Environment:** Visual Studio Code was used as the primary Integrated Development Environment (IDE).
- **Frameworks and Libraries:** The frontend was built using React and Tailwind CSS. The backend server was implemented with Node.js and Express. The core machine learning pipeline was developed in Python, leveraging TensorFlow for model building and training,

OpenCV for real-time video processing, and MediaPipe for keypoint extraction. The dataset was split into training (70%), validation (15%), and testing (15%) sets to ensure a robust and unbiased evaluation of the model's generalization capabilities.

### B. Baselines and Evaluation

The performance of the proposed ensemble CNN-BERT model was benchmarked against prior works in sign language recognition that utilize classical machine learning and earlier deep learning architectures, as reviewed in Section II. The evaluation focuses on the model's ability to correctly classify sign language gestures from video input. We employed the standard evaluation metrics of Accuracy, Precision, Recall, and F1-Score, as defined in Section IV-F, to provide a comprehensive assessment of the model's performance.

## V. RESULTS AND DISCUSSION

This section presents the empirical results of our proposed ensemble model and provides a detailed discussion of its performance. The findings demonstrate the model's high efficacy and its advancements over existing systems.

### A. Model Performance

The ensemble CNN-BERT model achieved state-of-the-art performance on the test dataset. A comprehensive summary of the evaluation metrics is presented in Table III. The model attained an overall accuracy of 98.7%, underscoring its ability to correctly classify a wide range of sign language gestures. Furthermore, it achieved a precision of 97.6% and a recall of 98.5%, culminating in an impressive F1-Score of 98.2%. This balance indicates that the model is not only accurate but also robust, minimizing both false positives and false negatives.

The training process showed excellent convergence, as illustrated by the learning curves. The training and validation loss decreased steadily over epochs, with a minimal gap between the two curves, suggesting that the model generalized well to unseen data without significant overfitting.

A visual analysis of the model's classification performance is provided by the confusion matrix. The matrix was highly diagonal, confirming that the vast majority of signs were classified correctly. Minor confusions were observed only between a few signs with very similar handshapes or motion trajectories (e.g., 'M' and 'N'), which is a common challenge in sign language recognition. Overall, the low off-diagonal values in the matrix corroborate the high accuracy reported in Table III.

**TABLE III: Overall Performance of the Ensemble CNN-BERT Model**

| Metric | Score (%) |
|---|---|
| Accuracy | 98.7 |
| Precision | 97.6 |
| Recall | 98.5 |
| F1-Score | 98.2 |

### B. Discussion

The exceptional performance of our model can be attributed to its hybrid architectural design, which synergistically combines the strengths of CNNs and Transformers.

- **Architectural Synergy:** The 3D-CNN component proved highly effective at extracting discriminative spatio-temporal features from the input video, capturing the subtle nuances of hand shape, orientation, and movement. This rich feature representation was crucial for the model's high discriminative power. The BERT component then processed this sequence of features, leveraging its bidirectional self-attention mechanism to understand the contextual relationships between signs. This allowed the model to interpret the grammatical flow of sign sequences, leading to more coherent and contextually appropriate translations than what is achievable with models that process signs in isolation.
- **Comparison with Prior Work: The** achieved accuracy of 98.7% significantly surpasses the performance of traditional vision-based systems that rely on hand-crafted features (Truong et al., 2016), (Tiku et al., 2020) and even surpasses many earlier deep learning approaches that used standalone CNNs or RNNs (Rao et al., 2018). Our results align with the trend reported by Camgoz, (2020) and Zhou et al., (2021) confirming that Transformer-based architectures are pivotal for advancing the state-of-the-art in sign language translation.
- **Impact of Dataset and Preprocessing:** The robustness of the model was greatly enhanced by the diversity of the training dataset. By incorporating data from various signers and including non-manual features, the model learned to generalize across different user styles and environmental conditions. The preprocessing pipeline, particularly the use of MediaPipe for accurate keypoint extraction, played a vital role in normalizing the input and allowing the model to focus on semantically meaningful information.
- **Practical Implications and Limitations:** The high accuracy and efficiency of the proposed system demonstrate its strong potential for real-world applications, such as integration into mobile devices for on-the-go communication support. However, like any machine learning system, its performance is contingent on the quality and scope of its training data. A key limitation remains the challenge of real-time performance under highly variable environmental conditions, such as poor lighting or occluded backgrounds. While our model is robust, extreme conditions can still impact recognition accuracy.

In summary, the results validate our hypothesis that an ensemble of CNNs and Transformers provides a powerful framework for accurate and context-aware sign language translation.

## VI. CONCLUSION AND FUTURE WORK

### A. Conclusion

In this study, we proposed and developed a novel ensemble learning-based system for bidirectional sign language translation, successfully bridging the communication gap for individuals with hearing or speech impairments. Our framework integrates a 3D Convolutional Neural Network (CNN) for robust spatio-temporal feature extraction with a Bidirectional Encoder Representations from Transformers (BERT) model to capture the complex contextual and grammatical nuances of sign language. The system demonstrated state-of-the-art performance, achieving an outstanding accuracy of 98.7% and an F1-Score of 98.2% on a diverse American Sign Language dataset.

The key contribution of this work lies in the synergistic fusion of spatial and sequential deep learning models, which enables a more holistic understanding of sign language than previously achieved. By designing the system to be bidirectional, computationally efficient, and accessible via standard hardware, we have laid the groundwork for a practical and scalable solution. The results confirm that our approach not only pushes the boundaries of automated sign language translation but also offers a tangible tool to foster greater inclusivity and accessibility in daily communication.

### B. Future Work

While our system has shown remarkable success, the field of automated sign language translation offers several promising avenues for future research. We identify the following directions to build upon our work:

- **Enhancing Continuous Sign Recognition:** Future work will focus on improving the model's ability to interpret long, continuous streams of conversational signing. This involves developing more advanced techniques to handle co-articulation, where the appearance of a sign is influenced by the signs preceding and following it.
- **Expanding Multilingual and Dialectal Support:** The current framework can be extended to include a wider array of sign languages and regional dialects. This re-quires the curation of larger, more diverse, and standardized datasets through community-driven data collection efforts.
- **Advanced Integration of Non-Manual Markers:** While our model incorporates facial and body cues, future research could explore more sophisticated methods for integrating non-manual markers, such as emotion recognition and gaze tracking, which are integral to conveying grammatical information and emotional tone in sign language.
- **Real-World Deployment and User Feedback:** A crucial next step is to deploy the system in real-world environments and gather feedback from the deaf and hard-of-hearing community. This will provide invaluable insights for refining the system's usability, accuracy, and overall user experience.
- **Integration with Augmented and Virtual Reality (AR/VR):** Exploring the integration of our translation engine with AR/VR technologies could create immersive and interactive platforms for sign language education and communication, offering novel ways to learn and practice signing.

By pursuing these research directions, we can further enhance the capabilities of automated translation systems and move closer to a world with seamless and universal communication.

## REFERENCES

Abou Haidar A., El Hajj J. & Amine H. (2019).” Glove-based sign language translator,” in 2019 1st International Conference on Communications, Signal Processing and their Applications (ICCSPA), 2019, pp. 1-5.

Abraham E., Nayak A., & Iqbal A. (2019). ”Real-time translation of Indian sign language using LSTM,” in 2019 Global Conference for Advancement in Technology (GCAT), 2019, pp. 1–5.

Bragg D. et al., (2019). Sign language recognition, generation, and translation: An interdisciplinary perspective, in The 21st International ACM SIGAC-CESS Conference on Computers and Accessibility, 2019, pp. 16–31.

Camgoz N., Koller O., Hadfield S., & Bowden R. (2020).” Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 10023-10033.

Fernandes L., Dalvi P., Junnarkar A., & Bansode M. (2020).” Convolutional neural network based bidirectional sign language translation system,” in 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, pp. 769–775.

Joshi A., Sierra H., & Arzuaga E. (2017).” American sign language translation using edge detection and cross correlation,” in 2017 IEEE Colombian Conference on Communications and Computing (COLCOM), 2017, pp. 1–6.

Karpathy, A., Toderici, G. Shetty S., Leung T., Sukthankar R. & Fei-Fei L. (2014)” Large-scale video classification with convolutional neural networks,” in IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.

Kumar S., Singh S., & Kumar A. (2018).” Real-time sign language recognition using a recurrent neural network,” in 2018 4th International Conference on Computing Communication and Automation (ICCCA), 2018, pp. 1-5.

Rao G., Kumar K., & Rao S. (2018).” Deep convolutional neural networks for sign language recognition,” in 2018 Conference on Signal Processing and Communication Engineering Systems (SPACES), 2018, pp. 194-197.

Sandler W. & Lillo-Martin D. (2006). Sign Language and Linguistic Universals. Cambridge: Cambridge University Press, 2006.

Shivashankara S. & Srinath D. (2018).” American Sign Language Recognition System: An Optimal Approach,” International Journal of Image, Graphics and Signal Processing, vol. 10, no. 8, pp. 1-10, 2018.

Truong V., Yang C., & Tran Q. (2016).” A translator for American sign language to text and speech,” in 2016 IEEE 5th Global Conference on Consumer Electronics (GCCE), 2016, pp. 1–2.

World Health Organization, “Deafness and hearing loss,” 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss.

Yin K., Read B., & Bowden R. (2020).” Sign Language Translation with Transformers,” in Asian Conference on Computer Vision, 2020, pp. 1-17.