IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 3, October 2025

Impact Factor: 7.67

Machine Learning Approaches for Hand Sign Recognition

Shreya Singh[1] and Prof. Sandeep NK[2]

Department of Master of Computer Applications[1]

Assistant Professor, Department of Master of Computer Applications[2]

Vidya Vikas Institute of Engineering and Technology, Mysore

Abstract: *This paper presents a deep learning-based system for real-time recognition of American Sign Language (ASL) gestures, including alphabets, digits, and common expressions. Using a modular pipeline—data acquisition, OpenCV preprocessing, MediaPipe Hands feature extraction, CNN classification in TensorFlow, and real-time text conversion—the system achieves over 92% accuracy across varied lighting and hand orientations at 30 FPS with a model under 1 MB. This approach addresses the shortage of interpreters and helps bridge communication gaps for the hearing-impaired community.*

Keywords: Sign Language Recognition, Deep Learning, Convolutional Neural Network, Computer Vision, MediaPipe, Real-time Systems, Assistive Technology, American Sign Language

I. INTRODUCTION

1.1 Background and Motivation

Communication represents a fundamental human need, yet millions of deaf and hard-of-hearing individuals face significant barriers in daily interactions. According to the World Health Organization, approximately 466 million people worldwide suffer from disabling hearing loss, with this number projected to reach 630 million by 2030 [1]. In India alone, an estimated 7 million deaf individuals are served by only 250 qualified sign language interpreters, creating an immense communication gap that affects education, employment, and social inclusion.

Sign languages are complete natural languages with their own grammar and syntax, distinct from spoken languages. Globally, between 138 and 300 sign languages exist, each with unique linguistic structures. American Sign Language (ASL), used by approximately 500,000 people in North America, serves as the focus of this research due to its well-documented vocabulary and widespread recognition.

1.2 Problem Statement

Traditional communication between deaf and hearing individuals has largely depended on human interpreters, who play a vital role in bridging the gap. However, interpreters are often scarce, expensive, and not always available when needed, which creates significant barriers to effective communication. In recent years, technological solutions have been introduced to address this challenge, but they still face several limitations. One major drawback is the limited vocabulary, as most existing systems can only recognize basic alphabets and numbers, making them inadequate for natural and expressive communication. Additionally, real-time performance remains a concern since the computational complexity of recognition models often hinders smooth live deployment. These systems are also highly sensitive to environmental factors, with performance degrading under varying lighting conditions or complex backgrounds. Moreover, many approaches depend on specialized sensors or high-end equipment, making them less accessible and practical for widespread use.

1.3 Research Objectives

This research aims to develop a robust and real-time sign language recognition system that effectively overcomes the limitations of existing solutions. To achieve this, a lightweight Convolutional Neural Network (CNN) architecture will be implemented to ensure efficient inference and faster processing, enabling smooth real-time performance. MediaPipe will be integrated for precise hand landmark detection, providing reliable tracking of hand movements without the need for specialized hardware. To enhance adaptability in diverse environments, comprehensive preprocessing techniques will be applied, ensuring robustness against variations in lighting and background conditions. Furthermore, the system will be designed as an extensible framework that supports multiple gesture types, allowing scalability beyond basic alphabets and numbers. Finally, a user-friendly interface will be developed to ensure practical deployment, making the solution accessible and easy to use for both deaf and hearing individuals.

1.4 Contributions

This paper presents a real-time ASL recognition system using CNNs and MediaPipe Hands, achieving over 92% accuracy at 30 FPS with a model under 1 MB, helping bridge communication gaps for the hearing-impaired.

II. LITERATURE REVIEW

2.1 Traditional Sign Language Recognition Approaches

Early sign language recognition systems primarily employed sensor-based methods and traditional computer vision techniques. Data gloves equipped with accelerometers, flex sensors, and gyroscopes provided accurate hand tracking but suffered from high cost, intrusiveness, and limited natural interaction [2]. Vision-based approaches emerged as non-intrusive alternatives, with Hidden Markov Models (HMMs) gaining popularity for temporal gesture recognition [3]. However, these methods required extensive manual feature engineering and struggled with complex gestures.

2.2 Machine Learning Methods in Gesture Recognition

Machine learning approaches significantly advanced the field of gesture recognition. Singha and Das [4] implemented Karhunen-Loeve Transforms (KLT) for hand gesture recognition, achieving 96% accuracy on ten gesture classes through coordinate system transformation and linear classification. Sharma et al. [5] combined k-Nearest Neighbors (k-NN) with Support Vector Machines (SVMs) using contour tracing descriptors, though achieving only 61% accuracy due to sensitivity to lighting variations and similar gestures.

2.3 Deep Learning Advances in Sign Language Recognition

Recent years have witnessed remarkable progress in deep learning applications for sign language recognition. Adewole et al. [6] developed a CNN-based system for Nigerian Sign Language, achieving 95% accuracy on static alphabets but lacking support for dynamic gestures. Bayrak et al. [7] introduced a novel approach using Complex-Valued Deep Neural Networks with Complex Zernike Moments, reaching 98.67% accuracy through advanced feature representation, though with increased computational complexity.

2.4 Real-time Systems and Integration Frameworks

The integration of sign language recognition with modern AI technologies has opened new possibilities. Chwesiuk and Popis [8] combined MediaPipe hand tracking with CNN models and Large Language Models, enabling intelligent dialogue systems for enhanced human-computer interaction. Verma et al. [9] developed a real-time system for ASL and Indian Sign Language (ISL) recognition, incorporating skin detection, background subtraction, and text-to-speech conversion for practical deployment.

2.5 Research Gaps and Opportunities

Despite these advancements, several challenges in sign language recognition remain unaddressed. Most existing systems continue to suffer from limited vocabulary coverage, restricting their ability to handle complex and natural conversations. Robustness to environmental variations is also insufficient, as performance often declines under changing lighting, backgrounds, or camera angles. Additionally, many approaches impose high computational requirements, which hinder their deployment on mobile and resource-constrained devices. The absence of standardized evaluation metrics and benchmark datasets further limits fair comparison and progress across different research efforts. Moreover, there is still

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 3, October 2025

Impact Factor: 7.67

limited support for continuous sign language recognition, with most systems focusing only on isolated gestures rather than fluid, sentence-level communication.

III. METHODOLOGY

3.1 System Architecture Overview

The proposed system employs a modular pipeline architecture consisting of six interconnected components that collectively ensure efficient and reliable real-time operation. The Data Acquisition Module is responsible for capturing live video streams using a standard webcam, eliminating the need for specialized hardware. The Preprocessing Module then enhances the captured images by applying normalization and image enhancement techniques to improve robustness under varying environmental conditions. Next, the Feature Extraction Module leverages MediaPipe to detect and process hand landmarks with high precision, providing accurate input features for classification. The Model Training Module focuses on implementing and optimizing a lightweight CNN architecture to achieve effective gesture recognition while maintaining computational efficiency. Once trained, the Real-time Inference Module handles live gesture classification and prediction, enabling smooth and responsive interaction. Finally, the User Interface Module presents recognition results and system status in a clear, user-friendly manner, ensuring practical usability for both deaf and hearing individuals.

3.2 Data Collection and Preparation

3.2.1 Dataset Composition

A comprehensive dataset was developed by integrating multiple sources to ensure diversity and robustness. First, a custom data collection effort was carried out, resulting in 8,000 images captured from 20 participants, thereby incorporating variations in signing styles, hand shapes, and backgrounds. In addition, the ASL Finger Spelling Dataset [10] contributed 5,000 annotated images, providing standardized samples for alphabet gestures. To further enhance variability and realism, the Microsoft ASL (MSASL) Dataset [11] was included, adding 2,000 video samples that capture dynamic signing conditions. The final curated dataset comprised a total of 15,000 samples spanning 36 classes, including 26 alphabets and 10 digits. Special attention was given to demographic diversity and signing variations to ensure that the dataset accurately reflects real-world conditions and improves the generalizability of the proposed system.

3.2.2 Data Augmentation Techniques

To enhance model robustness and minimize the risk of overfitting, extensive data augmentation techniques were applied to the dataset. Geometric transformations such as rotation within ±15°, scaling between 0.8x and 1.2x, and horizontal flipping were employed to account for variations in hand orientation and size. Photometric adjustments, including brightness variations of ±30%, contrast modifications of ±20%, and saturation changes, were introduced to simulate different lighting conditions. To improve resilience against visual distortions, noise was incorporated in the form of Gaussian noise with σ ranging from 0.01 to 0.05, as well as salt-and-pepper noise. Additionally, occlusion simulation was performed using random erasing, covering 10–20% of the image area, thereby training the model to recognize gestures even under partial visibility. These augmentation strategies collectively ensured that the model generalizes well to diverse real-world environments.

3.2.3 Dataset Partitioning

The dataset was strategically partitioned to ensure fair and reliable evaluation of the proposed system. A total of 70% (10,500 samples) was allocated to the training set to provide sufficient data for effective model learning. The validation set, comprising 15% (2,250 samples), was used during training to fine-tune hyperparameters and monitor model performance, thereby preventing overfitting. The remaining 15% (2,250 samples) formed the test set, which served as an unbiased benchmark for assessing the final system's accuracy and generalization capabilities. To maintain consistency and avoid class imbalance, stratified sampling was employed across all splits, ensuring that each class was proportionally represented in training, validation, and testing subsets. This partitioning strategy contributed to a balanced evaluation framework for robust performance analysis.

DOI: 10.48175/IJARSCT-29337

ISSN 2581-9429 IJARSCT

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

3.3 Preprocessing Pipeline

3.3.1 Image Enhancement

The preprocessing module employs a multi-stage enhancement pipeline designed to optimize image quality and feature extraction for reliable gesture recognition. The process begins with grayscale conversion, where RGB images are transformed into grayscale, reducing computational complexity while retaining critical structural features of the hand. Next, noise reduction is applied using adaptive Gaussian filtering with a 3×3 kernel to smooth out random variations, along with median filtering to effectively eliminate salt-and-pepper noise. To improve visibility of hand features under diverse lighting conditions, contrast enhancement is performed using Contrast Limited Adaptive Histogram Equalization (CLAHE), which enhances local contrast while avoiding over-amplification of noise. Finally, skin masking is implemented through segmentation in the HSV color space, enabling robust isolation of the hand region even under varying illumination and background conditions. This structured preprocessing pipeline ensures that the input fed into the feature extraction module is clean, consistent, and well-optimized for recognition.

3.3.2 Normalization and Standardization

In addition to the enhancement steps, the preprocessing module also incorporates normalization and standardization techniques to ensure consistency across the dataset. Size normalization is applied by resizing all images to a fixed resolution of 224×224 pixels, providing uniform input dimensions suitable for CNN-based architectures. To facilitate efficient learning and faster convergence, pixel normalization is performed by scaling pixel values to the range [0,1], thereby reducing intensity variations. Furthermore, data standardization is carried out by subtracting the mean and scaling by the standard deviation of the dataset, ensuring that the input distribution is centered and normalized. These steps collectively enhance model stability, improve training efficiency, and contribute to more accurate gesture recognition.

3.4 Feature Extraction using MediaPipe

3.4.1 Hand Landmark Detection

The MediaPipe Hands framework offers a detailed representation of hand geometry by providing 21 precise landmark points per hand (Figure 2), which correspond to key joints and features. Specifically, landmarks 0–4 represent the thumb, including its base joints and tip, while landmarks 5–8 correspond to the index finger components. The middle finger is captured by landmarks 9–12, the ring finger by landmarks 13–16, and the little finger by landmarks 17–20. The wrist base point is designated as landmark 0, serving as a reference for hand orientation. By tracking these landmarks, the system can accurately capture finger articulation and hand posture, forming the basis for effective feature extraction and gesture recognition.

3.4.2 Feature Vector Construction

The extracted hand landmarks are transformed into a comprehensive feature vector to effectively capture both spatial and temporal characteristics of gestures. Coordinate features consist of the normalized (x, y, z) positions of all 21 landmarks, providing precise spatial information. Distance features are computed as the Euclidean distances between selected landmark pairs, capturing relative positioning and hand shape. Angle features are derived from landmark triplets, representing joint angles that describe finger articulation and hand posture. Additionally, velocity features track temporal movement patterns across consecutive frames, enabling the system to capture dynamic aspects of gestures. Together, these features form a rich representation that supports accurate classification and real-time recognition.

3.5 CNN Architecture Design

3.5.1 Architectural Innovations

The CNN architecture was designed with several key considerations to optimize performance and generalization. A progressive filter increase from 32 to 64 and then 128 filters was employed to enable hierarchical feature learning, capturing both low-level and high-level patterns. Strategic dropout placement in the dense layers was incorporated to prevent overfitting and improve model robustness. Kernel size selection of 3×3 was chosen to extract fine-grained spatial features while maintaining computational efficiency. Additionally, a max pooling strategy was applied after convolutional

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

layers to introduce translation invariance, reduce spatial dimensions, and retain the most salient features, thereby enhancing overall recognition accuracy.

3.6 Training Methodology

3.6.1 Optimization Configuration

The training process was conducted using carefully selected parameters to ensure effective learning and model generalization. The Adam optimizer was employed with an initial learning rate of 0.001, providing adaptive gradient updates for faster convergence. Categorical cross-entropy was used as the loss function, suitable for multi-class gesture classification. Training was performed with a batch size of 32 samples per gradient update, balancing memory efficiency and gradient stability. The model was trained for 100 epochs, with early stopping applied based on validation loss to prevent overfitting. To further enhance generalization, L2 regularization with λ=0.01 and dropout layers were incorporated, mitigating overfitting while maintaining model performance.

3.6.3 Validation Strategy

To ensure robust and reliable evaluation, the model training incorporated several validation strategies. 5-fold cross-validation was employed to assess performance across multiple data partitions, reducing variance in the evaluation results. Stratified sampling was used within each fold to maintain consistent class distribution, ensuring that all gesture classes were proportionally represented in the validation sets. Model performance was measured using comprehensive metrics, including accuracy, precision, recall, F1-score, and confusion matrix analysis, providing a detailed understanding of the system's strengths, weaknesses, and class-wise prediction capabilities.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Experimental Setup

4.1.1 Hardware Configuration

The experimental setup for system development and testing utilized a mid-range computing platform to ensure practical deployment feasibility. The processor was an Intel Core i7-10750H running at 2.60 GHz, paired with 16 GB of DDR4 RAM to handle model training and real-time inference efficiently. A dedicated NVIDIA GeForce GTX 1650 Ti GPU with 4 GB memory was used to accelerate deep learning computations. For data acquisition and live testing, a Logitech C920 HD Pro webcam (1080p) captured high-quality video streams, providing reliable input for the gesture recognition pipeline. This configuration balanced computational performance with accessibility for real-world usage scenarios.

4.1.2 Software Environment

The software environment for system implementation was carefully selected to support deep learning and computer vision workflows. The experiments were conducted on Windows 11, providing a stable platform for development. The deep learning framework consisted of TensorFlow 2.8.0 and Keras 2.8.0, enabling efficient model building, training, and deployment. For computer vision tasks, OpenCV 4.5.5 was used for image processing operations, while MediaPipe 0.8.10 facilitated accurate hand landmark detection. The entire system was developed using Python 3.8.10, offering flexibility, ease of integration, and a wide range of libraries suitable for machine learning and real-time application development.

4.2 Performance Metrics

The system's performance was evaluated using a set of comprehensive classification metrics to provide a detailed assessment of its effectiveness. Accuracy was calculated as ((TP + TN) / (TP + TN + FP + FN)), representing the overall correctness of predictions. Precision, defined as (TP / (TP + FP)), measured the proportion of correctly predicted positive samples among all predicted positives. Recall, computed as (TP / (TP + FN)), assessed the model's ability to correctly identify all relevant samples. The F1-score, calculated as (2 \times (Precision \times Recall) / (Precision + Recall)), provided a balanced metric combining precision and recall. Additionally, inference time, measured in milliseconds per frame, quantified the system's real-time processing capability, ensuring practical usability for live gesture recognition.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

4.3 Qualitative Results and Case Studies

4.3.1 Successful Recognition Scenarios

The system demonstrated robust and reliable performance across a range of challenging conditions. Under variable lighting conditions, it maintained high accuracy, achieving 92.1% in daylight and 91.8% under artificial lighting, indicating strong illumination invariance. For hand orientation variations, the system successfully recognized gestures across a 0–180° rotation range, demonstrating effective spatial generalization. In scenarios involving partial occlusions, the model sustained 85% accuracy even when up to 30% of the hand was obscured, highlighting resilience to incomplete visibility. Furthermore, the system showed consistent performance across multiple skin tones, confirming its capability to handle demographic diversity and ensuring equitable recognition across different user groups.

4.3.2 Challenging Cases and Limitations

Certain scenarios posed challenges for gesture recognition, revealing areas for further improvement. Rapid gesture transitions occasionally caused brief misclassifications during fast signing, indicating sensitivity to temporal dynamics. Similar gesture pairs, such as ‘M’ and ‘N’, showed lower distinction, with recognition accuracy dropping to 78% due to subtle differences in finger positioning. Under extreme lighting conditions, particularly strong backlighting, overall performance decreased to 82%, reflecting limitations in illumination robustness. Additionally, complex and cluttered backgrounds led to approximately 15% reduction in accuracy, highlighting the need for enhanced segmentation and background-invariant feature extraction in challenging visual environments.

V. DISCUSSION

5.1 Performance Analysis

The proposed system achieves 92.3% overall accuracy, demonstrating significant improvement over traditional computer vision approaches while maintaining real-time performance. The integration of MediaPipe for feature extraction provides robust hand landmark detection, contributing to the system's environmental invariance. The lightweight CNN architecture ensures efficient operation without compromising recognition accuracy.

5.2 Comparative Advantages

The proposed system offers several key advantages over existing sign language recognition solutions. It delivers balanced performance, achieving high accuracy of 92.3% while maintaining real-time processing at 81.3 FPS, combining precision with speed. The system exhibits strong environmental robustness, sustaining consistent performance across varying lighting conditions and complex backgrounds. With hardware efficiency, it operates on modest computing resources, facilitating broader deployment without specialized equipment. Its extensible architecture allows for easy expansion of the gesture vocabulary, supporting scalability for future applications. Finally, the user-friendly interface provides intuitive feedback and system status, ensuring practical usability for both deaf and hearing users in real-world scenarios.

5.3 Limitations and Challenges

Despite its promising performance, the system has several limitations that warrant further attention. Vocabulary constraints currently limit recognition to 36 static gestures, excluding dynamic and sentence-level signs, which restricts expressiveness. Environmental sensitivity remains a challenge, as performance can degrade under extreme lighting or highly cluttered backgrounds. User dependency also affects accuracy, with variations in individual signing styles influencing recognition outcomes. The system’s reliance on a webcam introduces hardware constraints, limiting seamless deployment on mobile devices without integrated cameras of sufficient quality. Additionally, the focus on American Sign Language (ASL) reduces immediate applicability in other cultural or regional sign languages, highlighting the need for adaptation to broader linguistic contexts.

5.4 Practical Implications

The system holds significant potential for practical applications across multiple domains. In education, it can provide enhanced learning experiences for deaf students by enabling interactive and accessible content delivery. Within

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, October 2025
Impact Factor: 7.67

healthcare, the system can improve doctor-patient communication, ensuring accurate information exchange without the constant need for human interpreters. For public service accessibility, it can facilitate smoother interactions with government offices and commercial services, promoting inclusivity. In daily communication, the system bridges the gap between deaf and hearing individuals, supporting more natural and effective exchanges. Additionally, it serves as a valuable language learning tool for hearing individuals interested in learning sign language, fostering greater understanding and social integration.

VI. CONCLUSION AND FUTURE WORK

6.1 Conclusion

This research presents a comprehensive framework for real-time sign language recognition using deep learning and computer vision. The proposed system demonstrates that careful integration of modern technologies can create practical solutions for bridging communication gaps. The achieved 92.3% accuracy with real-time performance represents a significant advancement in making sign language recognition accessible and deployable in real-world scenarios.

The key successes of this work highlight its contributions to advancing sign language recognition technology. A robust preprocessing pipeline was developed, ensuring environmental invariance and consistent input quality under diverse lighting and background conditions. The system achieved effective integration of MediaPipe, enabling accurate and reliable hand landmark detection as the foundation for gesture recognition. A lightweight CNN architecture was designed, striking a balance between high accuracy and computational efficiency for real-time operation. The research culminated in the implementation of a complete real-time recognition system, capable of practical deployment. Finally, a comprehensive evaluation demonstrated the system's performance across various challenging scenarios, confirming its practical applicability and potential impact in real-world settings.

6.2 Future Research Directions

6.2.1 Technical Enhancements

Future enhancements of the system can focus on several directions to extend its capabilities. Dynamic gesture recognition can be achieved by integrating temporal models such as LSTM or Transformer networks, enabling the system to capture sequential patterns and recognize continuous signing. Multi-modal fusion offers the potential to combine hand gestures with facial expressions and body pose analysis, providing richer contextual understanding for more accurate communication interpretation. The exploration of advanced architectures, including attention mechanisms and capsule networks, could improve feature representation and classification performance. Additionally, transfer learning can facilitate adaptation to multiple sign languages by leveraging pre-trained models, reducing the need for extensive dataset collection and enabling broader global applicability.

6.2.2 Practical Extensions

Future work can further enhance the system's usability and scope through several practical improvements. Mobile deployment can be achieved by optimizing the model for smartphones and embedded devices, enabling on-the-go accessibility. Vocabulary expansion would allow support for the complete ASL lexicon and continuous signing, increasing expressiveness and real-world applicability. User personalization can be incorporated by developing adaptive models that learn individual signing styles, improving accuracy for diverse users. Finally, real-time translation can be implemented by integrating speech synthesis, enabling bidirectional communication between deaf and hearing individuals and facilitating seamless interaction.

6.2.3 Accessibility Improvements

Additional future directions focus on accessibility, scalability, and educational impact. Low-cost solutions can be developed to create affordable hardware, promoting widespread adoption in resource-constrained settings. Offline functionality would enable the system to operate entirely without internet connectivity, making it usable in areas with limited network access. Multi-user support could allow recognition of multiple signers simultaneously, enhancing usability in group interactions and collaborative environments. Furthermore, educational integration can be pursued by designing curriculum-aligned learning tools and applications, providing structured support for both deaf students and individuals learning sign language.

6.3 Societal Impact

The successful development and deployment of sign language recognition technology has profound societal implications. By reducing communication barriers, such systems promote inclusivity, enhance educational opportunities, and improve quality of life for deaf and hard-of-hearing individuals. As technology advances, the vision of seamless communication between deaf and hearing communities becomes increasingly attainable, representing a significant step toward a more inclusive and accessible society.

REFERENCES

[1] World Health Organization. (2021). Deafness and hearing loss. WHO Press.
[2] Starner, T., & Pentland, A. (1997). Real-time ASL recognition from video using HMM. Computer Vision and Image Understanding, 9(1), 227-243.
[3] Jeballi, M., et al. (2014). Extension of HMM for recognizing large vocabulary of sign language. International Journal of Artificial Intelligence & Applications, 4(2), 35-42.
[4] Singha, J., & Das, K. (2013). Hand gesture recognition based on Karhunen-Loeve transform. In Proceedings of MECON (pp. 365-371).
[5] Sharma, R., et al. (2013). Using a contour tracing descriptor to recognize single-handed sign language gestures. In Proceedings of WCE (Vol. II).
[6] Adewole, D. B., et al. (2024). Nigerian Sign Language recognition using deep learning. Federal University of Technology.
[7] Bayrak, S., et al. (2024). ASL recognition model using complex Zernike moments. Karadeniz Technical University.
[8] Chwesiuk, M., & Popis, P. (2024). ASL file conversion and integration with LLM services. Warsaw University.
[9] Verma, A., et al. (2024). Sign language recognizer using computer vision. HMR Institute of Technology
[10] Microsoft Research. (2020). MS-ASL dataset. Retrieved from https://github.com/microsoft/MS-ASL
[11] ASL Finger Spelling Dataset. (2019). Retrieved from https://www.kaggle.com/datasets
[12] Gonzalez, R. C., & Woods, R. E. (2018). Digital image processing (4th ed.). Pearson.

APPENDICES

Appendix A: Dataset Statistics
Detailed breakdown of dataset composition, participant demographics, and collection methodology.
Appendix B: Hyperparameter Configurations
Complete listing of all hyperparameters tested during model development and optimization.
Appendix C: Confusion Matrices
Detailed confusion matrices for each gesture category and overall system performance.
Appendix D: Ethical Considerations
Discussion of privacy, data protection, and ethical deployment considerations for sign language recognition systems