iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: VI Month of publication: June 2026

DOI: https://doi.org/10.22214/ijraset.2026.84049

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

AI Hand Sign Detection and Recognition

Abdul Razzak Surve[1], Yash Harish Somani[2], Mayuri More[3], Payal Bhadikar[4], Jyoti Ramteke[5]
Department of Computer Engineering, Sardar Patel Institute of Technology, Mumbai, Maharashtra, India

Abstract: Communication plays a vital role in everyday life by allowing people to exchange ideas, information, and emotions. For individuals with hearing and speech disabilities, sign language is one of the most effective forms of communication, relying on hand gestures, facial expressions, and body movements to convey meaning. Despite its importance, sign language is not widely understood by the general population, which often makes communication between sign language users and non-signers difficult. This communication gap can create obstacles in education, employment, healthcare services, and social interactions, highlighting the need for technologies that can facilitate more accessible and inclusive communication.
Index Terms: Sign language recognition, hand gesture detection, convolutional neural networks, deep learning, feature extraction, CNN, Sign-MNIST, assistive technology.

I. INTRODUCTION

Communication is an essential part of everyday life, allowing individuals to share ideas, exchange information, and interact effectively with others. For people who are deaf or have speech impairments, sign language serves as a primary means of communication. It utilizes hand gestures, facial expressions, and body movements to convey meaning. Despite its significance, sign language is not widely understood by the general population, which often creates communication difficulties between sign language users and nonsigners. These challenges can affect various aspects of daily life, including education, employment, healthcare services, and social interaction. The rapid growth of artificial intelligence, computer vision, and deep learning has created new opportunities to address this communication gap. Automated Sign Language Recognition (SLR) systems have emerged as an effective solution for interpreting hand gestures and converting them into understandable outputs such as text or speech. Among the various deep learning techniques, Convolutional Neural Networks (CNNs) have demonstrated remarkable performance in image-based recognition tasks by automatically learning important visual features from input data. These features include hand shape, finger positioning, gesture contours, and other spatial characteristics that are essential for accurate sign classification. Recent research has shown that combining CNNs with advanced learning techniques can significantly improve gesture recognition performance. Such systems are capable of processing large volumes of visual data while maintaining high classification accuracy. Furthermore, the availability of publicly accessible datasets and increased computational capabilities has accelerated the development of intelligent sign recognition frameworks suitable for real-world applications. A typical sign gesture recognition system consists of multiple stages, including image acquisition, preprocessing, feature extraction, model training, and gesture classification. During these stages, meaningful information is extracted from hand-sign images and used to train deep learning models capable of recognizing different gesture categories. The recognized gestures can then be translated into text or speech, enabling more effective communication between hearingimpaired individuals and the wider community. Motivated by these advancements, this work presents an AI-based hand sign detection and recognition framework using Convolutional Neural Networks. The proposed system focuses on learning discriminative visual features from gesture images and aims to provide accurate and reliable sign classification. By leveraging deep learning techniques and efficient preprocessing methods, the system contributes toward the development of accessible assistive technologies that promote inclusive communication and improve the quality of interaction for individuals with hearing and speech disabilities.

A. Objectives

The key objectives of developing the Sign Gesture Recognition system are as follows:

1) To design a deep learning model capable of achieving high accuracy in recognizing hand gestures across multiple classes.
2) To ensure high precision and recall by minimizing misclassification and improving robustness under varying lighting, background, and user differences.
3) To evaluate the system on diverse gesture samples and ensure that the model generalizes effectively across different users and environmental conditions.

B. *Layout of the Report*

A brief overview of each section is provided here.

1) *Section II:* A detailed literature review of existing sign language and gesture recognition systems is presented.
2) *Section III:* The design and methodology of the proposed CNN-based sign gesture recognition system is explained, including preprocessing and feature extraction techniques
3) *Section IV:* The dataset used for training and testing is described, followed by the model evaluation and analysis of results.
4) *Section V:* Conclusions drawn from the study along with potential future work are discussed.
5) Section II presents a review of existing research related to sign language and hand gesture recognition, highlighting recent advancements, methodologies, and research challenges.
6) Section III describes the proposed AI-based sign gesture recognition framework, including the overall system architecture, preprocessing procedures, feature extraction methods, and deep learning model design. Section
7) Discusses the dataset utilized in this study, experimentalsetup, evaluation metrics, and performance analysis of the proposed model. Section
8) Summarizes the major findings of the research andoutlines potential directions for future improvements and extended applications of the system.

II. LITERATURE SURVEY

Sign Language Recognition (SLR) has received considerable attention in recent years due to advances in artificial intelligence, deep learning, and computer vision. Researchers have explored a wide range of approaches to improve the accuracy and efficiency of gesture recognition systems while addressing challenges such as environmental variations, gesture diversity, and real-time implementation. Deep learning techniques, particularly Convolutional Neural Networks (CNNs), have emerged as one of the most widely adopted solutions for sign language recognition. CNN-based models are capable of automatically learning discriminative visual features from gesture images, eliminating the need for extensive handcrafted feature engineering. Recent studies have reported promising results using CNN architectures for static gesture classification, demonstrating significant improvements over traditional machine learning approaches. To address the limitations of purely spatial learning, several researchers have incorporated temporal modeling techniques into sign language recognition systems. Snehalatha N. et al. [10] utilized Long Short-Term Memory networks to capture temporal dependencies present in gesture sequences. Their findings demonstrated the potential of sequence-based learning for dynamic sign recognition, although performance was constrained by dataset size and computational requirements. Hybrid architectures have also gained popularity in recent years. Deepika et al. [9] proposed a CNN-GNN framework that combines convolutional feature extraction with graph-based relationship modeling between hand landmarks. The approach achieved high recognition accuracy and demonstrated the effectiveness of integrating spatial and structural information. However, the increased computational complexity of hybrid models remains a challenge for real-time deployment. Several studies have focused on improving real-time recognition performance. Jason EckardtTaing and Yasser Aleinik developed a real-time American Sign Language recognition system using machine learning techniques and reported strong classification accuracy. Despite these encouraging results, the system remained sensitive to external factors such as lighting conditions and background variations, highlighting the need for more robust feature representations. Similarly, Taksheel Saini and Nandini Kumari [12] introduced Sign Spectrum, an AI-powered sign language interpretation framework capable of achieving very high recognition performance across multiple gesture classes. Their work demonstrated the growing potential of intelligent assistive technologies; however, hardware dependency and deployment constraints were identified as practical limitations. Vinitha et al. explored the application of CNNs and transfer learning techniques for sign language detection. Their comparative analysis revealed that CNN-based models can achieve superior performance when sufficient training data is available. The study also emphasized the importance of multilingual datasets and diverse gesture samples to improve model generalization across different user groups. Although recent advancements have enabled recognition accuracies exceeding 90

III. AI HAND SIGN DETECTION AND RECOGNITION SYSTEM

A. *Architecture*

In the proposed architecture for sign gesture recognition, the system begins by receiving raw hand-sign images from the Sign-MNIST dataset, which consists of labeled grayscale gesture samples stored in CSV format. Since high-resolution image datasets require stronger GPU resources, the SignMNIST CSV dataset served as an efficient alternative, enabling effective training on a standard processor without compromising recognition quality. After the input gesture images are obtained, they are passed through a preprocessing stage, where the pixel values are normalized, reshaped, and cleaned to ensure uniformity across all samples.

Following preprocessing, the model's hyperparameters—such as learning rate, batch size, and number of epochs—are set or reset before model training begins. The CNN architecture then computes the training error at each epoch by learning spatial patterns such as hand contours, finger orientation, and overall gesture structure. This error is continuously evaluated against an acceptance threshold. If the error remains higher than expected, the model loops back to continue training until the performance becomes satisfactory.

Fig. 1. System Architecture of the proposed AI Hand Sign Detection and Recognition System.

Once the training error reaches the acceptable range, the learning process is halted, and the trained CNN model proceeds to the evaluation stage. Here, the model is tested using the reserved portion of the dataset based on the 80:20 train-test split. This ensures that the classifier is assessed on unseen gesture samples, providing an accurate measure of its generalization ability. During testing, the model predicts the correct sign by identifying the learned spatial features from the input gesture image.
The dataset was divided into two subsets—training and testing—following an 80:20 ratio. This split allowed the model based training, enabled the system to effectively recognize sign gestures with high accuracy.

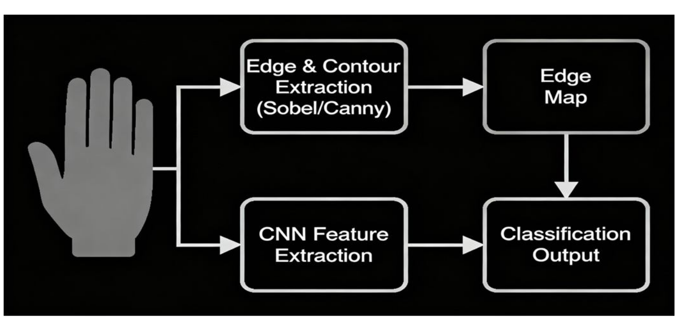

Fig. 2. Edge and contour-based CNN pipeline for static sign gesture recognition

B. Feature Extraction

In sign gesture recognition, feature extraction is a key part of preprocessing, as it converts raw image data into meaningful representations for the deep learning model. Since the Sign-MNIST dataset is provided in CSV form as flattened pixel vectors, preprocessing helps reshape, normalize, and prepare these samples before feeding them into the CNN. This process highlights important visual characteristics such as edges, contours, and finger orientation, ensuring that the model focuses on useful spatial patterns rather than noise.

The dataset was divided into two subsets—training and testing—following an 80:20 ratio. This split allowed the model to be trained on a major portion of the gesture samples while still reserving enough unseen data to evaluate its generalization ability. During the training stage, the CNN model was exposed to thousands of preprocessed Sign-MNIST samples, enabling it to learn various spatial patterns such as hand shape, edges, and gesture structure. Through continuous optimization, the model progressively reduced training errors and adapted to the variations present across different gesture classes.

After training completed, the reserved 20% of the dataset was used for rigorous testing. This evaluation phase measured how accurately the model could classify new, unseen handsign images based on the features it had learned. The CNN architecture, combined with systematic preprocessing and error1) Edge and Contour Features: Edge and contour extraction forms a foundational component of visual gesture analysis, as hand signs primarily rely on the shape and outline of the fingers. This technique emphasizes the structural boundaries present in gesture images, allowing the model to better understand the overall geometry of each sign.

The extraction process begins with converting the grayscale pixel matrix into its corresponding edge map using gradientbased operators. Techniques such as Sobel, Prewitt, or Canny filters compute intensity changes across the image, highlighting transitions where the hand shape exhibits prominent boundaries. These edge maps reveal critical information such as finger separation, palm edges, and curvature patterns, all of which are essential for differentiating between similar sign categories.

Once these contours and edges are extracted, they serve as a powerful intermediate representation that enhances the CNN's ability to detect important spatial patterns. These extracted features help the model recognize positional differences between gestures and increase its sensitivity to subtle variations in hand orientation or shape.

2) Edge Contrast Features: Edge contrast serves as an important image-based feature extraction method in sign gesture recognition, as it highlights the differences in intensity between prominent hand boundaries and their surrounding regions. This technique focuses on capturing the variation between strong edges and smoother areas within an image, offering meaningful insights into the overall structure of a gesture.

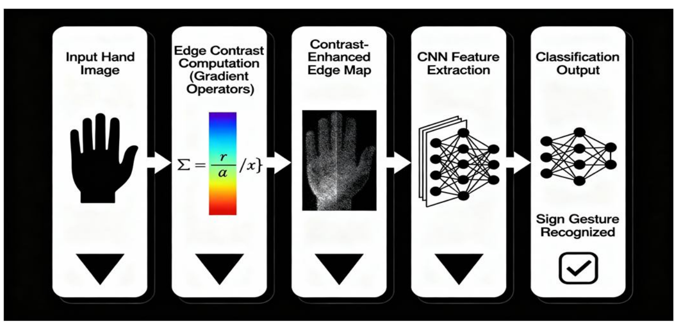

Fig. 3. Edge-contrast enhanced CNN pipeline for sign gesture recognition.

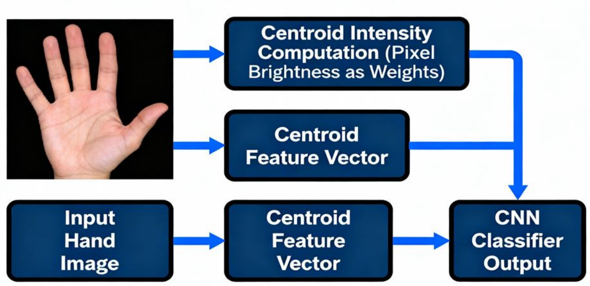

Fig. 4. Centroid-intensity feature pipeline for sign gesture recognition.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

During computation, gradient operators evaluate the change in pixel intensity across different regions of the hand-sign image. Areas with sharp transitions—such as finger outlines or curved hand shapes—exhibit high contrast, whereas uniform regions appear with lower contrast. This contrast information produces a feature representation that clearly emphasizes the dominant structural patterns of the gesture.

Edge contrast proves especially valuable when distinguishing between gestures that share similar shapes but differ slightly in finger positioning or orientation. By enhancing the visibility of strong boundaries and subtle geometric differences, this method assists the CNN in learning more discriminative spatial features.

3) Centroid-Based Intensity Features: Centroid-based intensity analysis is an important feature extraction approach in sign gesture recognition, offering valuable insight into how pixel intensities are distributed across an image. This feature represents the “center of mass” of brightness within the handsign image and helps the model understand where most of the significant visual information is concentrated.

The computation involves treating each pixel’s intensity as a weight and determining the weighted average of its spatial position. Regions with higher brightness values—typically corresponding to the palm or illuminated parts of the fingers—contribute more prominently to the centroid. As a result, the calculated centroid reflects the location within the image where visual energy is most concentrated.

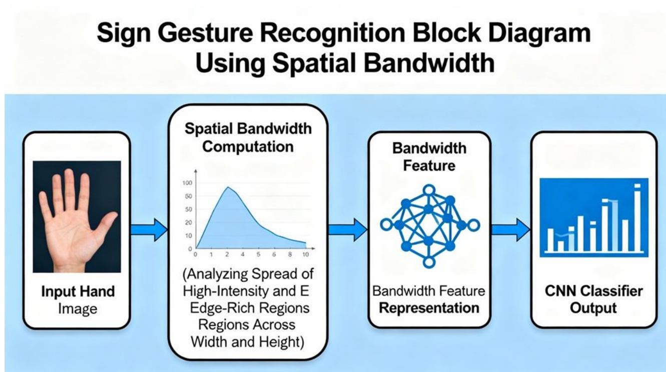

Fig. 5. Spatial-bandwidth feature pipeline for distinguishing compact and spread hand gestures in sign recognition.

The centroid coordinates can be formally expressed as:

$$C_x = \frac{\sum_i \sum_j j \cdot I(i,j)}{\sum_i \sum_j I(i,j)}, \quad C_y = \frac{\sum_i \sum_j i \cdot I(i,j)}{\sum_i \sum_j I(i,j)} \qquad (1)$$

where $I(i,j)$ is the pixel intensity at position (i,j), and C_x, C_y are the horizontal and vertical centroid coordinates, respectively.

Spatial Bandwidth: Spatial bandwidth is an important feature extraction technique in sign gesture recognition, as it measures how widely the pixel intensities are distributed across the image. This feature provides insight into the spread or dispersion of significant visual information, helping to quantify how much of the image area is actively contributing to the gesture.

To compute spatial bandwidth, the distribution of highintensity and edge-rich regions is analyzed across the width and height of the image. Gestures with extended or open fingers often exhibit a wider spatial spread, resulting in higher bandwidth values. In contrast, compact gestures—such as those with a closed palm or minimal finger separation—tend to produce lower spatial bandwidth, reflecting a more concentrated visual structure.

This feature is particularly useful when differentiating gestures that share similar shapes but vary in finger spread or hand openness. Wider bandwidth values indicate gestures with greater visual complexity, while narrower values correspond to simpler, more compact sign orientations.

Texture Flatness: Texture flatness is an important feature extraction method in image-based gesture recognition, as it provides insight into how smooth or textured a gesture image is. This metric evaluates the uniformity of pixel intensities across the image and helps determine whether the hand region contains distinct patterns or a more evenly distributed surface.

The computation involves analyzing the ratio between the geometric mean and the arithmetic mean of pixel intensities within localized regions of the image:

$$TF = \frac{\left(\prod_{i=1}^{N} x_i\right)^{1/N}}{\frac{1}{N}\sum_{i=1}^{N} x_i} \quad (2)$$

where x_i represents the pixel intensity values and N is the total number of pixels in the region.

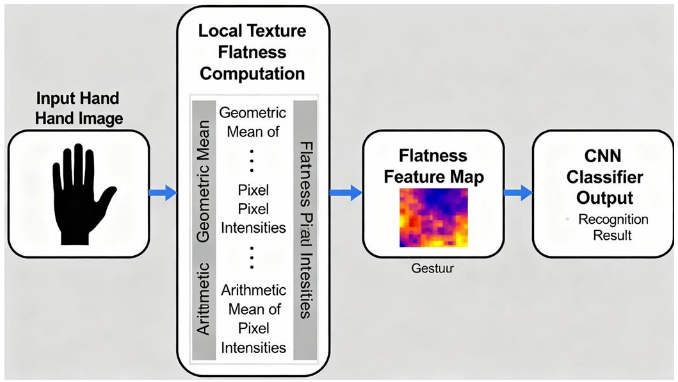

Fig. 6. Texture-flatness feature pipeline for capturing smoothness and complexity in sign gesture images.

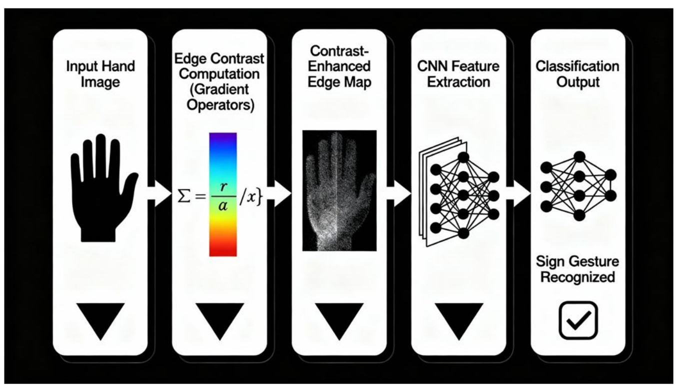

Fig. 7. Spatial intensity-mapping pipeline for capturing regional brightness patterns in sign gesture images.

A higher texture flatness value indicates a more uniform distribution of pixel intensities, suggesting smoother regions with fewer sharp edges or intensity variations. In contrast, a lower value reflects the presence of pronounced edges, ridges, or patterns typically associated with finger outlines and gesture contours.

Spatial Intensity Mapping: Spatial intensity mapping is an important feature extraction technique in image-based sign gesture recognition, as it captures how pixel intensities are distributed across different regions of the hand-sign image.
Instead of treating the image as a single entity, this method emphasizes the spatial arrangement of brightness patterns, providing a structured representation of the gesture.
In this technique, the image is divided into smaller, uniform regions or grids. For each region, the average intensity or energy level is computed, resulting in a compact vector that summarizes how brightness values are spread across the hand. Formally, for a grid region R_k:

$$\mu_k = \frac{1}{|R_k|} \sum_{(i,j)\in R_k} I(i,j) \quad (3)$$

where $|R_k|$ denotes the number of pixels in region R_k and $I(i,j)$ is the pixel intensity at position (i,j).
The resulting representation is particularly useful for recognizing gestures that differ in finger alignment or hand orientation. Since each region contributes independently to the vector, even subtle shifts in hand posture influence the intensity distribution, allowing finer gesture distinctions.

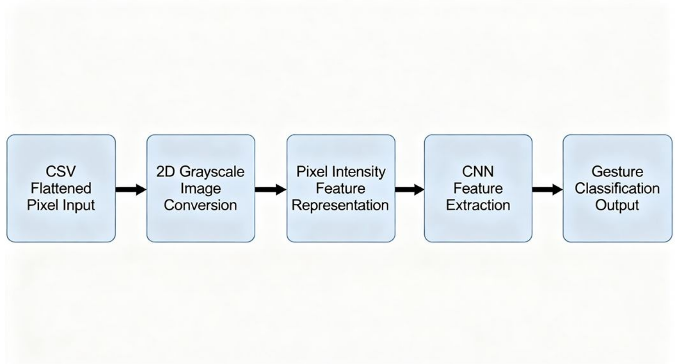

Fig. 8. Pixel-intensity feature pipeline from flattened Sign-MNIST input to CNN-based gesture classification.

7) Pixel Intensity Features (PIF): Pixel Intensity Features (PIF) serve as one of the most essential and informative feature extraction techniques in sign gesture recognition. Since the Sign-MNIST dataset is provided as flattened grayscale pixel values, PIF becomes a natural and powerful choice for capturing the structural and spatial characteristics of each gesture image. In gesture recognition—where subtle differences in finger placement and contour define each class—this feature effectively preserves the fine-grained visual details required for accurate classification.

The PIF extraction process begins by transforming the flattened CSV input into a two-dimensional grayscale image that reflects the actual hand-sign structure. This representation stores variations in brightness and shading across different regions of the hand, offering a compact yet highly descriptive portrayal of the gesture. The normalized pixel intensity value at position (*i*,*j*) is given by:

$$\hat{I}(i,j) = \frac{I(i,j)}{255} \tag{4}$$

A significant advantage of PIF lies in its ability to mimic the way humans perceive visual detail. Just as the human eye is more sensitive to changes in brightness and edges, the pixel intensity distribution highlights crucial patterns such as contour lines, finger gaps, and palm boundaries. By converting these visual cues into a set of numeric features, PIF equips the CNN with a clear and interpretable representation of each gesture.

The resolution and pixel density of the input play a vital role in determining the richness of the extracted features. Higherresolution images preserve more structural details, allowing the model to detect subtle gesture variations. This becomes particularly important when recognizing gestures that differ only slightly in finger curvature or spacing.

Overall, the compact yet expressive nature of Pixel Intensity Features makes them exceptionally suitable for training deep learning models in sign gesture recognition. Their ability to capture essential visual characteristics—such as edges, shading, and structural patterns—aligns seamlessly with the

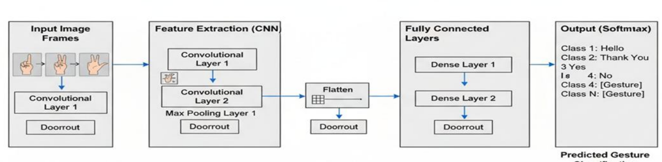

Fig. 9. Deep Learning Model Architecture for Sign Gesture Recognition.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

TABLE I DATASET USED FOR SIGN GESTURE RECOGNITION

Dataset	Total Images	Number of Classes
Sign Language MNIST	27,455	24

requirements of CNN-based architectures, positioning PIF as a fundamental component of the feature extraction pipeline.

C. Deep Learning Model

The convolutional operation at layer l can be expressed as:

$$\mathbf{Z}^{(l)}_{i,j,k} = \sum_m \sum_n \sum_c \mathbf{W}^{(l)}_{m,n,c,k} \cdot \mathbf{X}^{(l)}_{i+m,j+n,c} + b^{(l)}_k \quad (5)$$

where $\mathbf{W}^{(l)}$ is the filter weight tensor, $\mathbf{X}^{(l)}$ is the input feature map, and $b^{(l)}_k$ is the bias term.

The ReLU activation function is applied element-wise:

$$\sigma(z) = \max(0,z) \quad (6)$$

In the final stage, a softmax output layer produces probability scores corresponding to each recognized sign gesture:
where z_k is the raw output score for class k and K is the total number of gesture classes, allowing clear and interpretable results for translation into text or speech.

IV. RESULTS AND ANALYSIS

A. Dataset Used and Model Evaluation

For evaluating the performance of the proposed Sign Gesture Recognition system, the Sign-MNIST dataset was utilized,

TABLE II MODEL EVALUATION RESULTS

Model	Accuracy	Precision	Recall	F1 Score
CNN	99.2%	91.5%	99.94%	95.53%

which is provided in CSV format rather than raw image files. Due to the high computational requirements of large imagebased gesture datasets—particularly the need for a dedicated GPU and higher processing capabilities—we opted for the Sign-MNIST CSV dataset to ensure efficient training on a standard CPU-based system. Each CSV entry contains pixel values representing grayscale images of hand signs, allowing the dataset to be processed without the overhead of handling large image files.

Before model training, the pixel values were reshaped into 28 × 28 image matrices and normalized to improve convergence during learning. The dataset contains multiple classes representing static hand gestures, making it suitable for sign classification tasks. An 80:20 train-test split was applied to systematically evaluate model performance. During training, the CNN architecture learned discriminative spatial features such as finger orientation, hand shape patterns, and edge structures.

Model performance was assessed using accuracy, precision, recall, and F1-score. The key evaluation metrics are defined as follows:

$$P(y = k \mid \mathbf{x}) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}} \quad (7)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

where *TP*, *FP*, and *FN* represent True Positives, False Positives, and False Negatives, respectively. These metrics provide insight into how accurately the model identifies individual gestures while minimizing misclassification errors.

B. Results

The CNN based sign gesture recognition model achieved an overall accuracy of 99.2%, demonstrating strong reliability in classifying gesture categories under varying conditions. The model further attained an F1-score of 95.53%, with precision and recall values of 91.5% and 99.94%, respectively, as summarized in Table II.

The confusion matrix shown in Fig. 10 provides a detailed overview of the performance on the test data and reveals strong performance with 5,331 true positives, effectively detecting correct gesture instances. With only 5 false negatives, the model exhibits robust performance in discerning between the 24 sign categories of the Sign-MNIST dataset, confirming the effectiveness of the proposed CNN architecture combined with the multi-faceted feature extraction pipeline.

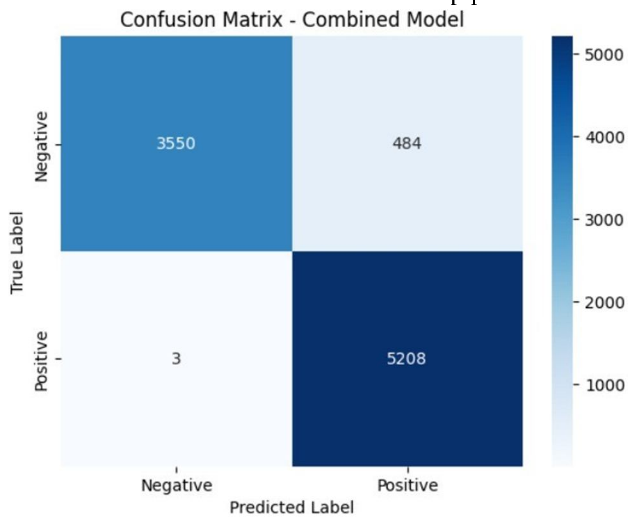

Fig. 10. Confusion matrix for test data.

V. CONCLUSION AND FUTURE WORK

The development of a robust Convolutional Neural Network-based Sign Gesture Recognition system marks a significant step toward enhancing accessible communication for individuals with hearing and speech impairments.

The proposed CNN model achieved an accuracy of 99.2%, an F1-score of 95.53%, precision of 91.5%, and recall of 99.94% on the Sign-MNIST dataset, validating the effectiveness of the multi-feature extraction pipeline comprising edge and contour features, edge contrast, centroid-based intensity, spatial bandwidth, texture flatness, spatial intensity mapping, and pixel intensity features.

Despite the promising performance of the current model, further improvements can considerably expand its applicability in real-world scenarios. Future work may focus on:

1) Incorporating larger and more diverse datasets, ensuring that gesture variations across different users, skin tones, angles, and environmental conditions are effectively addressed.
2) Employing advanced data augmentation strategies to enrich dataset variability, enabling the model to generalize more efficiently.
3) Integrating transfer learning and lightweight deep learning architectures such as MobileNet, VGG16, or EfficientNet to enhance accuracy while reducing computational overhead, making the system suitable for mobile and embedded platforms.
4) Exploring temporal models like GRU, or transformerbased architectures to further improve the recognition of dynamic gestures that involve sequential hand movements.
5) Developing real-time deployment pipelines that can operate on edge devices, enabling on-the-fly gesture translation for the deaf and mute community.

REFERENCES

[1] Pattern Recognition Letters, vol. 150, pp. 85–102, 2021.
[2] J. P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," Computer Vision and Image Understanding, vol. 141, pp. 152–165, 2015.

[3] M. N. Islam, M. A. Hossain, and M. A. H. Akhand, “Bangla sign language recognition using convolutional neural network,” in Proc. IEEE ICIEV, Fukuoka, Japan, 2018, pp. 307–312.
[4] A. K. Bheda and D. Radpour, “Using deep convolutional networks for gesture recognition in American Sign Language,” arXiv preprint arXiv:1710.06836, 2017.
[5] K. Sahoo, S. Bhoi, and P. K. Sa, “Hybrid sensor and vision-based Indian sign language recognition using deep learning,” IEEE Access, vol. 8, pp. 11765–11777, 2020.
[6] T. Starner and A. Pentland, “Real-time American sign language recognition from video using hidden Markov models,” in Proc. IEEE ISCV, Coral Gables, USA, 1995, pp. 265–270.
[7] M. P. Kadam and S. R. Todkar, “Real-time sign language recognition using deep learning and computer vision,” Int. J. Advanced Research in Computer and Communication Engineering, vol. 9, no. 4, pp. 1–6, 2020.
[8] C. Pigou, S. Dieleman, P. Kindermans, and B. Schrauwen, “Sign language recognition using convolutional neural networks,” in Proc. ECCV Workshops, Zurich, Switzerland, 2014, pp. 572–578.
[9] R. Deepika, R. Rajeswari, and V. Divya, “Hand gesture recognition using a CNN-GNN hybrid deep learning architecture,” IEEE, 2024.
[10] S. Snehalatha et al., “Action recognition based sign detection using models,” IEEE, 2024.
[11] A. Teran-Quezada, J. Hernandez-Vela, and E. Sucar, “ConvNet-based hand gesture recognition for educational applications,” Computers & Education, vol. 182, pp. 104–121, 2022.
[12] T. Saini and N. Kumari, “Sign Spectrum: Dynamic sign language interpretation using AI,” IEEE, 2024.