International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

SMART SIGN-TO-VOICE AND TEXT CONVERSION SYSTEM FOR INDIAN SIGN LANGUAGE USING ESP32-CAM AND MACHINE LEARNING

Bonala Bhavyasri, N Sukumar
Department of Electronics and Communication Engineering,
Jayamukhi Institute of Technological Sciences, Narsampet, Warangal, Telangana, India

ABSTRACT
Communication between individuals who use Indian Sign Language (ISL) and people who do not understand sign language remains a significant accessibility challenge. This paper presents a low-cost, wireless, camera-based system for converting selected Indian Sign Language gestures into text and audible speech. The proposed system uses an ESP32-CAM for image acquisition and Wi-Fi-based transmission to a PC or server, where OpenCV performs image preprocessing and a machine-learning/deep-learning classifier identifies the corresponding gesture. The recognized gesture is mapped to a textual label and subsequently converted into speech using a Text-to-Speech (TTS) engine. The system integrates embedded vision, wireless communication, computer vision, machine learning, and speech synthesis into a unified assistive communication framework. The implementation uses a predefined gesture vocabulary and collects gesture samples through the ESP32-CAM. The supplied implementation defines ten gesture classes, namely HELLO, YES, NO, HELP, STOP, THANK YOU, PLEASE, SORRY, WELCOME, and I LOVE YOU, with 150 samples specified for each class. The recognition pipeline employs MediaPipe hand detection and 63-dimensional hand-landmark features before neural-network classification. The resulting architecture separates computationally intensive recognition from the embedded image-acquisition unit, enabling a compact and portable design. Functional experiments demonstrate the complete workflow from gesture acquisition and wireless transmission to gesture recognition, text generation, and speech output. The proposed system provides a practical foundation for affordable assistive communication and can be extended toward continuous ISL recognition, multilingual translation, and fully embedded edge-AI implementation.

Keywords— Indian Sign Language, ISL, ESP32-CAM, hand gesture recognition, machine learning, computer vision, MediaPipe, OpenCV, text-to-speech, assistive technology, embedded vision.

I. INTRODUCTION

Communication is a fundamental requirement for social interaction, education, healthcare, employment, and independent participation in society. For individuals who primarily communicate through sign language, interaction with people who are unfamiliar with sign language can become difficult, particularly in public environments where professional interpreters may not be immediately available. Indian Sign Language (ISL) is a complete visual-gestural language used by the Deaf community in India and has its own vocabulary and linguistic structure. The Indian Sign Language Research and Training Centre (ISLRTC), Government of India, has developed an extensive ISL Dictionary containing thousands of sign representations, emphasizing the importance of ISL in communication, education, and accessibility [1]. The project considered in this work is motivated by this communication gap and aims to provide an automated mechanism for converting selected ISL gestures into text and speech.

Computer vision has emerged as an important technology for automated sign-language recognition because it allows gestures to be captured without requiring users to wear specialized sensing devices. Earlier systems commonly relied on image segmentation, handcrafted features, and conventional machine-learning classifiers. More recent approaches increasingly employ convolutional neural networks (CNNs), recurrent networks, attention mechanisms, graph-based representations, and Transformer architectures to learn discriminative spatial and temporal representations from sign-language data [2]–[5].

For Indian Sign Language specifically, vision-based recognition has demonstrated promising performance. Gangrade and Bharti investigated vision-based ISL hand-gesture recognition using a CNN and reported the effectiveness of image-based recognition for ISL gestures [2]. The study demonstrates that CNN-based models can learn visual characteristics of hand configurations without requiring wearable sensors. A separate study by Ashwanth *et al.* also investigated webcam-based ISL recognition using CNN-based processing and reported 93.5% test accuracy [3].

The availability of large-scale ISL datasets has further accelerated research. Sridhar *et al.* introduced the INCLUDE dataset, containing thousands of videos representing isolated ISL signs and providing a resource for machine-learning-based recognition research [4]. The dataset contains 4,287 videos and approximately 0.27 million frames covering 263 word signs.

More recently, research has moved beyond isolated gesture recognition toward continuous sign-language translation. Joshi *et al.* introduced ISLTranslate, a continuous Indian Sign Language translation dataset containing approximately 31,000 ISL-English sentence and phrase pairs and benchmarked Transformer-based translation models on the dataset [5]. Patra *et al.* subsequently introduced a large-scale isolated ISL dataset containing 40,033 videos covering 2,002 commonly used words and proposed a Hierarchical Windowed Graph Attention Network for skeleton-based sign recognition [6].

Although these developments have substantially improved recognition capabilities, practical deployment remains challenging. High-performing recognition systems often depend on large datasets, powerful processing hardware, or complex temporal models. Furthermore, performance can be affected by illumination, background clutter, camera position, hand orientation, signer variability, occlusion, and motion. Recent surveys have identified computational efficiency, dataset diversity, real-world robustness, and the transition from isolated recognition to continuous translation as important open issues in sign-language recognition [7], [8].

The present work therefore focuses on a practical and economical assistive communication architecture rather than attempting unrestricted continuous sign-language translation. The proposed system uses an ESP32-CAM as the image acquisition and wireless communication unit. Captured gesture images are transmitted through Wi-Fi to a PC or server, where OpenCV performs preprocessing and the trained machine-learning/deep-learning model performs gesture classification. The recognized gesture is then mapped to text and supplied to a TTS engine to generate audible speech. The complete processing chain is:

This architecture combines embedded vision, wireless communication, machine learning, and speech synthesis within a single assistive communication framework.

The main contributions of this work are:

1. Integration of OpenCV-based preprocessing with machine-learning-based gesture classification.
2. Development of an end-to-end gesture-to-text and gesture-to-voice communication pipeline.
3. Use of a lightweight hand-landmark representation for efficient gesture classification.
4. Development of a practical architecture in which computationally intensive recognition is performed on a PC/server while the ESP32-CAM performs image acquisition and wireless transmission.

II. RELATED WORK

Sign-language recognition is an interdisciplinary research area involving computer vision, machine learning, artificial intelligence, natural-language processing, and assistive technology. Recognition systems can broadly be categorized into isolated-sign recognition and continuous-sign recognition. Isolated recognition determines the class of an individual sign, whereas continuous recognition processes a sequence of signs and attempts to infer the corresponding linguistic meaning.Early vision-based systems generally employed image acquisition followed by preprocessing, hand segmentation, handcrafted feature extraction, and conventional classification. Singha and Das investigated recognition of Indian Sign Language from live video and reported a success rate of 96.25% for 24 ISL alphabets using preprocessing and feature-based recognition [9]. Such approaches

demonstrated the feasibility of camera-based recognition without requiring users to wear gloves or other sensing devices. However, handcrafted features can be sensitive to illumination, background, hand orientation, scale, and signer variability. The limitations of conventional feature engineering motivated the adoption of machine-learning and deep-learning techniques. CNNs are particularly suitable for image-based gesture recognition because they can automatically learn spatial features associated with different hand configurations.

Gangrade and Bharti investigated vision-based hand-gesture recognition for ISL using CNNs. Their work demonstrated that webcam images can be processed using a CNN-based recognition framework for ISL gestures [2]. Ashwanth *et al.* subsequently presented another vision-based ISL gesture-recognition system using webcam images and CNN-based processing, reporting 93.5% test accuracy [3]. CNN-based approaches provide an important advantage over handcrafted feature methods because spatial representations can be learned directly from image data. However, image-based CNN models may require significant computational resources, particularly when the input image resolution and model size increase. In addition, a conventional image classifier primarily captures spatial characteristics and does not inherently model temporal relationships between consecutive signs. The proposed system addresses this practical consideration by separating the embedded image acquisition stage from the computationally intensive recognition stage. The ESP32-CAM captures and transmits images, while a PC/server performs the machine-learning computation.

Deep learning has become a dominant approach for sign-language recognition. Rastgoo *et al.* presented a comprehensive survey covering vision-based sign-language recognition methods, datasets, modalities, and deep-learning architectures [7]. Their analysis highlights the importance of hand shape, movement, body information, facial information, and temporal context.

More recent approaches combine CNN-based spatial feature extraction with recurrent or attention-based temporal modelling. Recurrent neural networks, LSTM and GRU architectures can model temporal dependencies in sign sequences, while attention and Transformer-based architectures can capture relationships across longer sequences. A recent review of deep-learning approaches for video-based sign-language recognition further emphasizes the transition toward temporal modelling, attention mechanisms, hybrid architectures, and Transformer-based methods [8]. These developments provide improved recognition capabilities but increase computational and data requirements.

The availability of suitable datasets is a major factor influencing the development of reliable ISL recognition systems. Compared with extensively studied sign languages such as American Sign Language, ISL has historically had fewer large-scale publicly available datasets. Sridhar *et al.* introduced INCLUDE, a large-scale ISL dataset containing 4,287 videos and approximately 0.27 million frames covering 263 word signs from multiple categories [4]. Joshi *et al.* introduced ISLTranslate to address continuous ISL translation. The dataset contains approximately 31,000 ISL-English sentence or phrase pairs and was designed specifically for research beyond isolated signs [5]. Patra *et al.* introduced a larger isolated ISL resource containing 40,033 videos covering 2,002 commonly used words recorded from 20 adult signers. Their work also introduced a Hierarchical Windowed Graph Attention Network that represents the human upper-body skeleton as a graph [6]. These datasets demonstrate the evolution of ISL recognition research from small controlled vocabularies toward larger vocabularies and more realistic signing conditions.

Recognition of individual signs represents only one stage toward natural sign-language communication. In practical conversations, users generate sequences of signs rather than isolated gestures. Continuous recognition therefore requires modelling temporal relationships and linguistic context.

ISLTranslate is particularly relevant because it focuses on continuous ISL-English translation using sentence and phrase-level data [5]. Patra *et al.* also demonstrated the potential of skeleton-based modelling for isolated ISL recognition with a large vocabulary [6]. These studies indicate that future ISL communication systems should move beyond isolated classification toward sequence recognition and

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

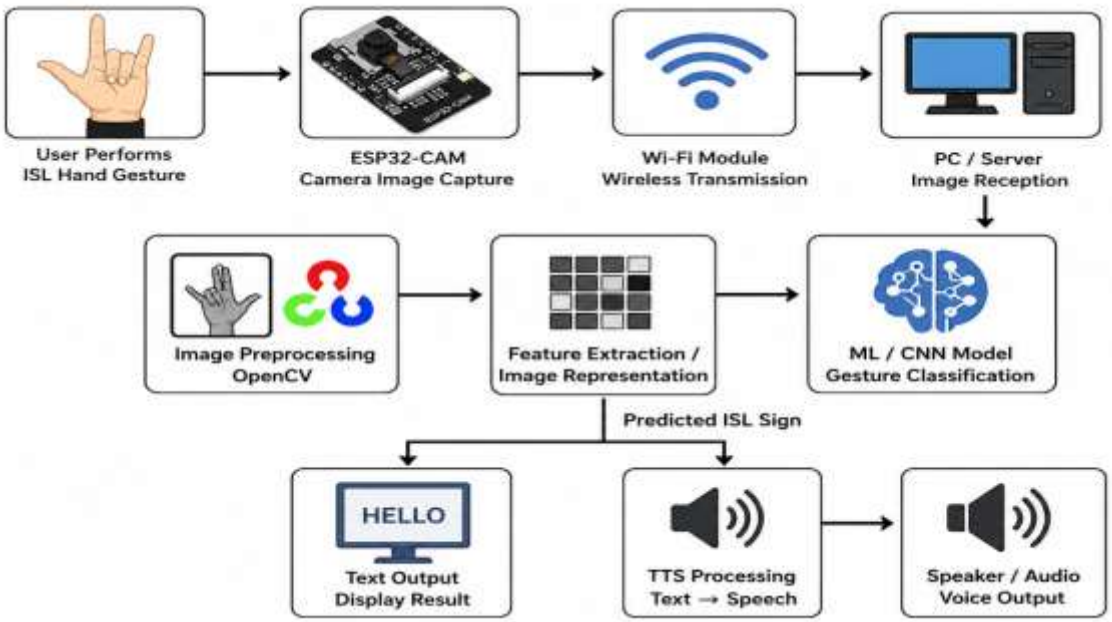
translation. However, such systems require substantially larger datasets and more computationally intensive architectures than the predefined-gesture system presented in this work.

Recognition accuracy alone does not guarantee practical communication. An assistive system should provide an output that can be understood by individuals who do not know sign language. Consequently, sign-to-text and sign-to-speech conversion are important extensions of sign recognition.
The proposed system maps the predicted gesture class to a textual label and subsequently passes the text to a TTS engine. The source implementation identifies gTTS, eSpeak, and pyttsx3 as possible speech-generation components. This two-stage approach provides both visual and auditory outputs. Text can be displayed to the communication partner, while speech enables immediate auditory interpretation.

Sign-language recognition can be implemented using vision-based or sensor-based techniques. Sensor-based approaches may use instrumented gloves, inertial measurement units, flex sensors, or other wearable devices. Although such systems can capture detailed hand-motion information, they require additional hardware to be worn by the user. Camera-based systems provide a more natural interaction mechanism because users can perform gestures without wearing specialized equipment. However, camera-based recognition is affected by illumination, background clutter, occlusion, viewpoint, and camera quality. The proposed system follows the vision-based approach using an ESP32-CAM. The camera and Wi-Fi interface are integrated into a compact embedded module, while computationally demanding recognition is performed externally. This division provides a practical compromise between portability and computational capability.

III. PROPOSED SYSTEM

The proposed system consists of six major stages ISL gesture acquisition using ESP32-CAM, Wireless transmission through Wi-Fi, Image preprocessing using OpenCV, Hand feature extraction and gesture classification, Gesture-to-text conversion and Text-to-speech conversion. The ESP32-CAM operates as the front-end image acquisition device. It captures the user's gesture and provides the image stream through Wi-Fi. A PC or server receives the image stream and performs the computationally intensive processing.



International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Figure 1 Block Diagram of proposed system

IV. METHODOLOGY

The ESP32-CAM is initialized and connected to a Wi-Fi network. The camera continuously captures images of the user's hand gesture. The captured frames are made available to the PC/server through the ESP32-CAM stream. The source implementation uses an ESP32-CAM stream URL and OpenCV's video-capture interface to access the stream. The received frame is processed before recognition. The implementation resizes the frame to 640 × 480 pixels, applies mild Gaussian filtering, and converts the frame from BGR to RGB representation for MediaPipe processing. The preprocessing stage reduces the influence of image noise and ensures that the frame is provided in a consistent format for subsequent hand detection.

MediaPipe Hands is employed for hand detection. The implementation is configured for one hand, with minimum detection and tracking confidence values of 0.70. This stage identifies the hand region and provides landmark information that can be used for classification. Instead of supplying the complete image directly to the classifier, the implementation extracts 21 hand landmarks. Each landmark contains x, y, and z coordinates, producing a 63-element feature vector. This representation substantially reduces the dimensionality of the classification input while retaining the geometric structure of the hand.

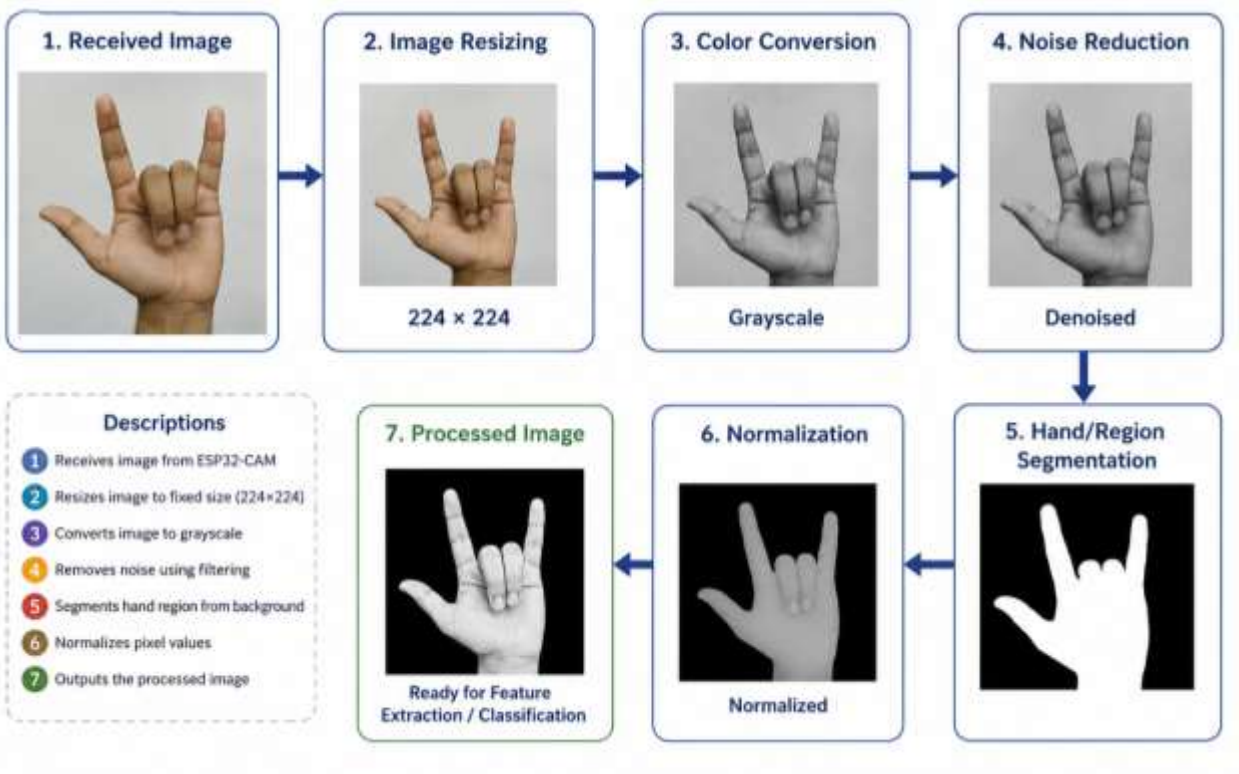


Figure 2 Image processing pipeline

The system collects gesture samples directly through the ESP32-CAM. For each gesture category, the system detects the hand, extracts the 63 features, and stores them as NumPy files. The implementation

specifies 150 samples per gesture. Therefore, if all ten defined gesture classes are populated, the intended dataset size is 1,500 samples. This is the configured collection size, not a verified final dataset size.

The extracted feature vectors are used to train the recognition model. Label encoding converts gesture names into numerical class identifiers, while the classification model learns the relationship between the hand-landmark representation and the gesture class. The implementation specifies 40 training epochs, a batch size of 32, and a classification confidence threshold of 0.80.

After classification, the predicted class is mapped to its corresponding gesture label. The resulting label is displayed as text. This provides an immediate visual representation of the user's intended communication.

The recognized text is passed to a TTS engine. The implementation provides pyttsx3 support and the project document also identifies gTTS and eSpeak as possible alternatives. The generated speech is played through a speaker, allowing a person unfamiliar with ISL to hear the recognized message.

I. Complete Operating Procedure

The complete system operates according to the following sequence:

- Initialize ESP32-CAM.
- Establish Wi-Fi communication.
- Capture the ISL gesture.
- Transmit the image to the processing unit.
- Receive the image.
- Preprocess the image using OpenCV.
- Detect the hand.
- Extract hand features.
- Classify the gesture.
- Map the predicted class to text.
- Display the recognized text.
- Pass the text to the TTS engine.
- Generate speech.
- Play the speech.

1. Repeat the process for the next gesture.

V. EXPERIMENTAL SETUP

The experimental system consists of the ESP32-CAM, FTDI programmer, PC/server, display, speaker, power supply, and Wi-Fi network. The PC/server performs image processing and machine-learning classification, while the ESP32-CAM provides image acquisition and wireless communication. The software environment comprises the Arduino/ESP32 development environment, Python, OpenCV, the ML/CNN model, TTS software, and Wi-Fi communication.

TABLE I: FUNCTIONAL PERFORMANCE OF THE PROPOSED SYSTEM

Parameter	Observed Function
Gesture acquisition	ISL hand gesture captured using ESP32-CAM
Wireless communication	Image transmitted through Wi-Fi
Image processing	OpenCV-based processing performed
Hand detection/extraction	Relevant hand representation obtained
Gesture recognition	ML/CNN model predicts ISL class
Text conversion	Predicted gesture converted into text
Text display	Recognized text presented to user
Speech conversion	Text converted using TTS
Audio output	Speech reproduced through speaker
Continuous operation	Recognition process can be repeated

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

The ESP32-CAM successfully forms the front-end acquisition component of the proposed architecture. The captured gesture image is transmitted to the PC/server through Wi-Fi, eliminating the need for a physical communication cable during normal operation. This arrangement improves flexibility in positioning the camera and processing unit and contributes to the portability of the system.

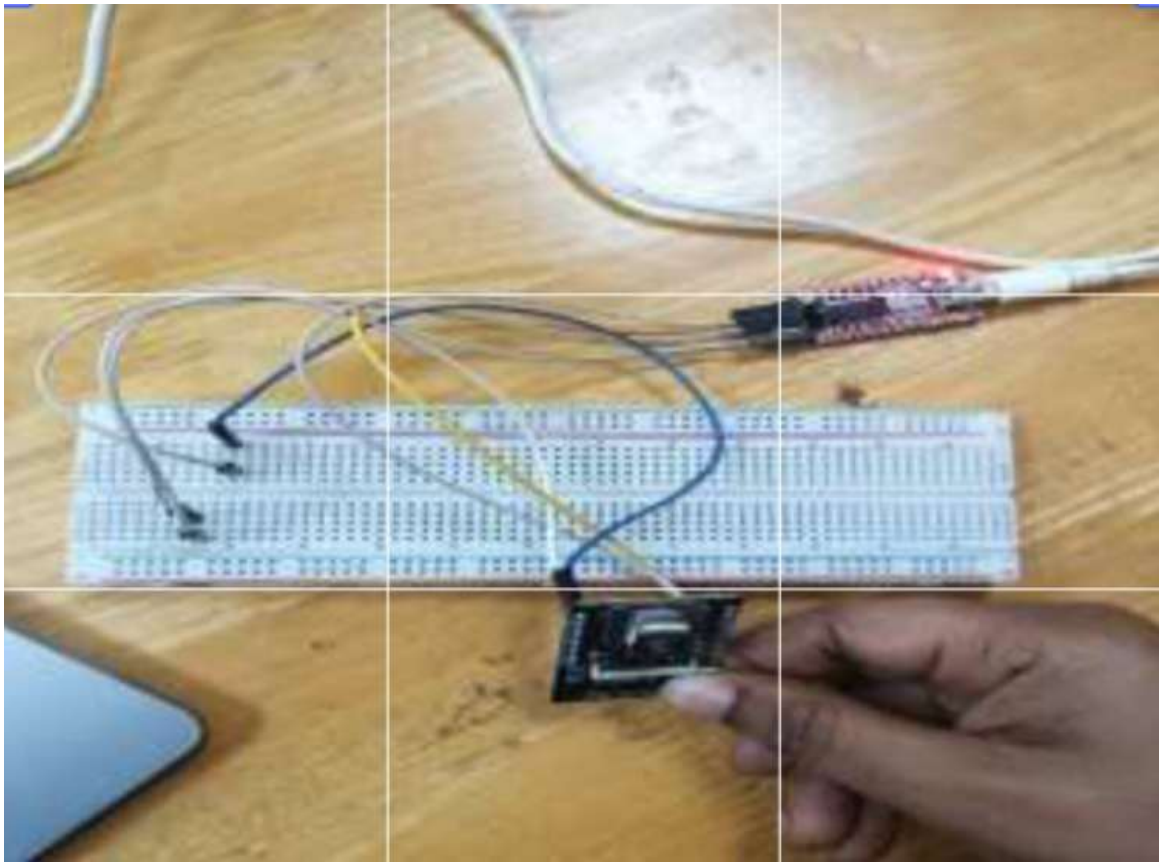


Figure 3 experimental output

The processed image is supplied to the machine-learning classification stage, where the corresponding gesture class is predicted. The predicted class is then mapped to the associated text label.
The supplied project document confirms the intended sequence from OpenCV processing through ML/CNN classification and subsequent text generation. One of the important characteristics of the proposed system is that the output is not restricted to classification labels. The predicted gesture is converted into readable text and then supplied to a TTS engine. The text output provides a visual communication channel, whereas speech output provides an auditory channel for people who do not understand ISL. The proposed system demonstrates the feasibility of combining embedded vision and machine learning for assistive communication. Unlike a conventional image-classification application, the system integrates image acquisition, wireless transmission, recognition, text generation, and speech synthesis into a single operational pipeline. The use of ESP32-CAM provides a compact front-end device containing both camera and wireless connectivity. The computational workload is shifted to a PC/server, which avoids imposing the full machine-learning workload on the embedded camera module. This architecture represents a practical trade-off between portability and computational capability.

The use of hand landmarks provides another important implementation characteristic. The supplied code detects 21 landmarks and generates a 63-dimensional feature vector from the x, y, and z coordinates. This representation can reduce the amount of information that must be processed compared with directly classifying full-resolution images.

VI. CONCLUSION

This paper presented a low-cost and wireless sign-to-voice and text conversion system for selected Indian Sign Language gestures using an ESP32-CAM and machine-learning-based recognition. The system combines embedded image acquisition, Wi-Fi communication, OpenCV preprocessing, hand detection, feature extraction, gesture classification, text generation, and Text-to-Speech conversion. The ESP32-CAM provides a compact front-end platform for capturing and transmitting hand gestures, while a PC/server performs the computationally intensive recognition process. The predicted gesture is mapped to text and subsequently converted into audible speech, providing both visual and auditory communication outputs. The implementation defines ten gesture categories and specifies 150 samples per gesture. It also employs MediaPipe hand detection and a 63-dimensional representation obtained from 21 hand landmarks. The functional experiments demonstrate the complete communication pipeline from gesture acquisition to wireless transmission, recognition, text output, and voice generation. The principal contribution of the work is therefore the integration of embedded vision and machine learning into an accessible communication framework rather than a claim of state-of-the-art recognition accuracy. The architecture provides a practical foundation for future development involving larger ISL vocabularies, continuous sign recognition, multilingual translation, multimodal information, and edge-AI deployment.

REFERENCES

[1] Indian Sign Language Research and Training Centre (ISLRTC), Government of India, *ISL Dictionary*, Department of Empowerment of Persons with Disabilities, Ministry of Social Justice and Empowerment.

[2] J. Gangrade and J. Bharti, "Vision-based Hand Gesture Recognition for Indian Sign Language Using Convolution Neural Network," *IETE Journal of Research*, vol. 69, no. 2, pp. 723–732, 2023, doi: 10.1080/03772063.2020.1838342.

[3] B. Ashwanth, S. B. Ventrapragada, S. R. Prodduturi, J. R. Depa, and K. V. Sharma, "Vision-based Hand Gesture Recognition for Indian Sign Language Using Convolution Neural Network," *International Journal of Computer Engineering in Research Trends*, vol. 10, no. 1, pp. 1–9, 2023.

[4] A. Sridhar, R. G. Ganesan, P. Kumar, and M. M. Khapra, "INCLUDE: A Large Scale Dataset for Indian Sign Language Recognition," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1366–1375, doi: 10.1145/3394171.3413528.

[5] A. Joshi, S. Agrawal, and A. Modi, "ISLTranslate: Dataset for Translating Indian Sign Language," in *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 10466–10475, 2023, doi: 10.18653/v1/2023.findings-acl.665.

[6] S. Patra, A. Maitra, M. Tiwari, K. Kumaran, S. Prabhu, S. Punyeshwarananda, and S. Samanta, "Hierarchical Windowed Graph Attention Network and a Large Scale Dataset for Isolated Indian Sign Language Recognition," arXiv:2407.14224, 2024.

[7] R. Rastgoo, K. Kiani, and S. Escalera, "Sign Language Recognition: A Deep Survey," *Expert Systems with Applications*, vol. 164, Art. no. 113794, 2021, doi: 10.1016/j.eswa.2020.113794.

[8] "A Comprehensive Review of Deep Learning Approaches for Video-Based Sign Language Recognition: Datasets, Challenges and Insights," *Multimodal Technologies and Interaction*, vol. 10, no. 6, Art. no. 58, 2026.

[9] J. Singha and K. Das, "Recognition of Indian Sign Language in Live Video," *International Journal of Computer Applications*, vol. 70, no. 19, pp. 17–22, 2013, doi: 10.5120/12174-7306.

[10] S. K. Sharma *et al.*, "Survey on Sign Language Recognition in Context of Vision-Based and Deep Learning," *Measurement: Sensors*, vol. 23, Art. no. 100385, 2022.

[11] "Real Time Indian Sign Language Recognition System," *Materials Today: Proceedings*, vol. 58, Part 1, pp. 504–508, 2022, doi: 10.1016/j.matpr.2022.03.011.
[12] K. Suri and R. Gupta, "Convolutional Neural Network Array for Sign Language Recognition Using Wearable IMUs," 2020.
[13] M. Kalinowski and B. Kostek, "Machine Learning in Sign Language: A Comprehensive Analysis and Trend Survey," *Computer Speech & Language*, 2026, Art. no. 100895.