



A MACHINE LEARNING BASED REAL-TIME SIGN LANGUAGE RECOGNITION SYSTEM WITH SPEECH CONVERSION

C. Fairlin Gold Teena¹, Dr. J. Evangelin Deva Sheela²

Department of Computer Applications, Sarah Tucker College, Tirunelveli-7.

ABSTRACT :

Sign language provides an important means of communication for people with hearing and speech disabilities. However, communication between sign language users and people who do not understand sign language can be challenging. This paper presents a real-time sign language recognition system using computer vision and machine learning techniques. The proposed system uses a webcam to capture hand gestures and *MediaPipe* to detect 21 hand landmarks from the left and right hands. The extracted landmark coordinates are used as numerical features for classification. A dataset containing 4,000 samples of 20 static word-level gestures was prepared for the system. Three machine learning algorithms, namely *K-Nearest Neighbors (KNN)*, *Support Vector Machine (SVM)*, and *Random Forest (RF)*, were implemented and evaluated using Accuracy, Precision, Recall, and F1-Score. The experimental results show that SVM achieved the highest offline accuracy of 96.36%. In real-time testing, SVM achieved an accuracy of 96.5%. The recognized gestures are displayed as text and converted into speech, providing an accessible communication interface. The proposed system demonstrates the potential of machine learning and computer vision for real-time sign language communication assistance.

Keywords: Sign Language Recognition, Machine Learning, MediaPipe, Hand Landmark Detection, KNN, SVM, Random Forest, Computer Vision, Speech Conversion, Real-Time Recognition

1. INTRODUCTION

Sign language is an important means of communication for people with hearing and speech disabilities. It uses hand gestures and movements to express words, ideas, and messages. However, communication can become difficult when the other person does not understand sign language. Therefore, an automated sign language recognition system can help reduce this communication barrier by converting hand gestures into understandable text and speech.

Recent developments in computer vision and machine learning have provided effective techniques for recognizing hand gestures from images and video. Hand landmark detection can represent the structure and position of the hand using numerical coordinates. In this work, *MediaPipe* is used to detect 21 hand landmarks from the captured hand gestures. The extracted landmark coordinates are then provided as features to machine learning algorithms for gesture classification.

This study presents a real-time machine learning-based sign language recognition system that uses a webcam to recognize 20 static word-level gestures. The system uses a dataset of 4,000 samples and evaluates *K-Nearest Neighbors (KNN)*, *Support Vector Machine (SVM)*, and *Random Forest* algorithms. The recognized gesture is displayed as text and converted into speech, providing a simple and accessible communication interface.

The objective of this work is to develop a real-time, user-friendly sign language recognition system that can accurately recognize predefined gestures and provide both text and speech output for communication assistance.

2. LITERATURE REVIEW

Sign language recognition has been studied using computer vision and machine learning techniques. Existing approaches commonly use hand landmark detection to represent hand gestures as numerical features. *MediaPipe* provides 21 hand landmarks that can be used for gesture classification, making landmark-based recognition suitable for real-time applications.

A recent study applied *MediaPipe* with multiple machine learning algorithms for American Sign Language recognition. The study used X and Y coordinates of 21 hand landmarks and evaluated classifiers including *KNN*, *SVM*, *Decision Tree*, and *Random Forest*. *Random Forest* achieved an accuracy of 98.83% on its alphabet-level dataset, demonstrating the effectiveness of *MediaPipe* landmark features for sign language recognition.

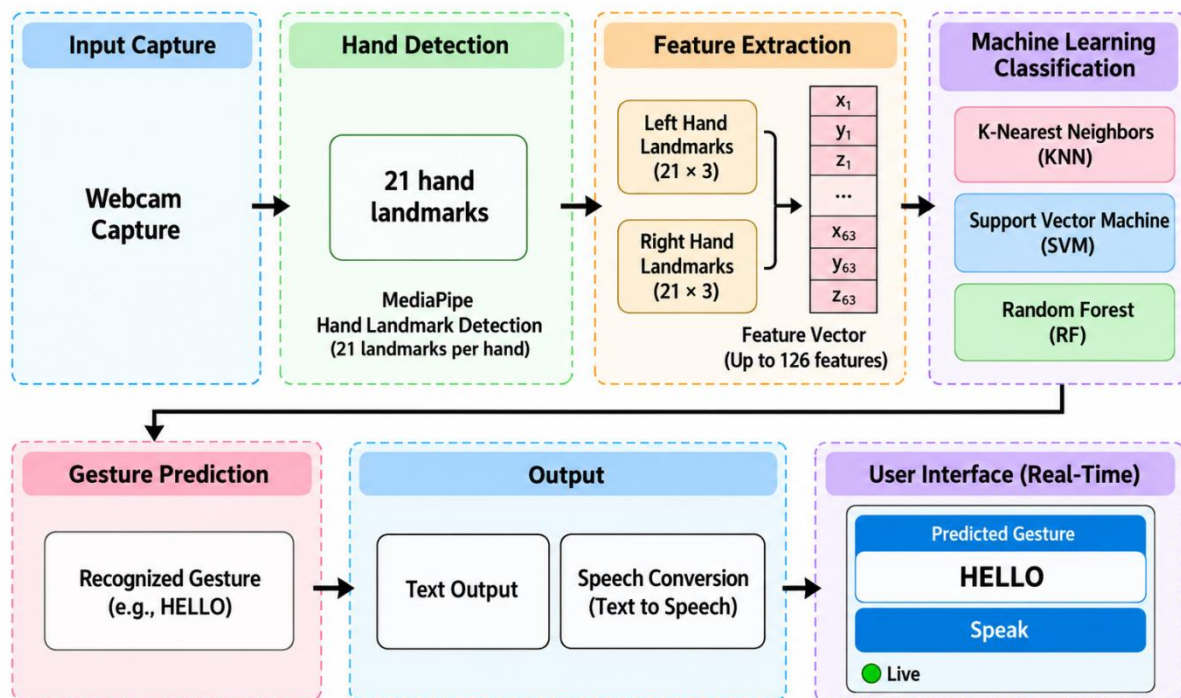
3. PROPOSED SYSTEM

The proposed system is a *real-time machine learning-based sign language recognition system* designed to recognize predefined static word-level hand gestures through a webcam. The system aims to reduce the communication barrier between sign language users and people who may not understand sign language. A webcam captures the user's hand gesture in real time, and the captured video frames are processed using computer vision techniques. The system is designed to recognize *20 predefined gestures*, including words such as HELLO, HELP, THANK YOU, YES, NO, FOOD, and WATER. For hand detection and landmark extraction, the system uses *MediaPipe*. It identifies *21 landmarks for each detected hand*, and each landmark contains *X, Y, and Z coordinate values*. Since the system supports both left and right hands, the feature representation contains up to $21 \times 3 \times 2 = 126$ numerical features. These extracted features represent the position and structure of the user's hands and are provided as input to the machine learning classification models. The system uses *K-Nearest Neighbors (KNN)*, *Support Vector Machine (SVM)*, and *Random Forest* for gesture classification. After classification, the predicted gesture is displayed as *text* on the user interface and is also converted into *speech*. This provides both visual and audio feedback to the user. The system includes real-time webcam processing, hand landmark detection, gesture prediction, text display, and speech conversion as its major functional components.

4. METHODOLOGY

The methodology of the proposed sign language recognition system consists of several stages, including *dataset preparation*, *hand detection*, *landmark extraction*, *feature extraction*, *machine learning model training*, *gesture classification*, and *speech conversion*. The system is designed to process hand gestures captured through a webcam and provide the predicted gesture as text and speech.

Figure 1 – System Architecture



4.1 Dataset Description

A self-collected dataset containing 4,000 samples of 20 static word-level sign language gestures is used for developing the system. The dataset includes gestures such as HELLO, HELP, THANK YOU, YES, NO, FOOD, WATER, DOCTOR, FRIEND, LOVE, and other predefined words. Both left-hand and right-hand gestures are included, allowing the system to handle gestures performed using one or both hands.

4.2 Data Collection and Preprocessing

The gesture samples are captured using a webcam under different hand positions and orientations. Each captured sample is processed to identify the hands and extract their landmark coordinates. The extracted numerical values are stored in a structured dataset for machine learning. Missing hand information is represented appropriately so that the prepared feature data can be consistently provided to the classification models.

4.3 Hand Landmark Detection Using MediaPipe

MediaPipe is used to detect and track the user's hands from the webcam frames. For each detected hand, 21 landmarks are identified. Each landmark contains *X*, *Y*, and *Z* coordinates, which represent the spatial position of the corresponding point on the hand. These landmarks provide a compact numerical representation of the hand gesture for subsequent classification.

4.4 Feature Extraction

Feature extraction converts the detected hand landmarks into numerical input features for machine learning. Each hand provides $21 \text{ landmarks} \times 3 \text{ coordinates} = 63 \text{ features}$. For two hands, the system can use $21 \times 3 \times 2 = 126 \text{ features}$. These features are used to represent the position and structure of the hands during gesture recognition.

4.5 Machine Learning Models

Three supervised machine learning algorithms are used for gesture classification:

- *Support Vector Machine (SVM)*: Classifies gestures by finding a suitable decision boundary between different gesture classes.
- *Random Forest*: Combines multiple decision trees to produce the final gesture classification.
- *K-Nearest Neighbors (KNN)*: Classifies a gesture based on the similarity between the input features and nearby training samples.

4.6 Model Training and Testing

The extracted landmark dataset is divided into training and testing data. The machine learning models are trained using the prepared gesture features and their corresponding labels. The trained models are then tested using unseen samples to evaluate their ability to correctly recognize the predefined gestures.

4.7 Performance Evaluation

The performance of the proposed sign language recognition system is evaluated using *Accuracy*, *Precision*, *Recall*, and *F1-Score*. These metrics are used to measure how effectively the machine learning models classify the 20 predefined sign language gestures.

Accuracy measures the percentage of correctly predicted gestures among all tested samples. *Precision* measures how accurately the model predicts a particular gesture, while *Recall* measures the ability of the model to correctly identify actual gesture samples. *F1-Score* provides a balanced measure of Precision and Recall.

In addition to these metrics, a *confusion matrix* is used to identify correctly classified and misclassified gestures. The models are also evaluated using real-time webcam testing to verify their practical performance. In the real-time evaluation, 600 tests were performed across 20 gestures and three algorithms, with SVM achieving 96.5% accuracy, KNN 86.0%, and Random Forest 36.5%

5. SYSTEM IMPLEMENTATION

5.1 Real-Time Webcam Module

The real-time webcam module captures video frames continuously from the user's webcam. Each frame is processed to detect the hand gestures performed by the user. OpenCV is used for webcam access and frame processing. The captured frames are then passed to the MediaPipe hand landmark detection module for further processing.

5.2 Gesture Recognition Module

The gesture recognition module identifies the hand gesture from the extracted landmark features. MediaPipe detects the hand landmarks, and the corresponding *X*, *Y*, and *Z* coordinates are converted into a feature vector. The trained machine learning models—KNN, SVM, and Random Forest—use these features to predict the corresponding sign language gesture. The predicted gesture is displayed in real time.

5.3 Text Output Module

The text output module displays the recognized gesture as text on the user interface. For example, when the system recognizes the corresponding hand gesture, the predicted word such as "HELLO", "HELP", or "THANK YOU" is displayed on the screen. This provides immediate visual feedback to the user.

5.4 Speech Conversion Module

The speech conversion module converts the recognized text into spoken audio. After a gesture is classified, the predicted word is passed to the text-to-speech component. The system then generates speech corresponding to the recognized gesture, allowing the output to be heard by another person.

5.5 User Interface

The user interface provides an integrated environment for real-time sign language recognition. It displays the webcam feed, detected hand landmarks, predicted gesture, and speech-related controls. The interface allows the user to observe the recognition process and receive both text and speech output. The system is implemented using *CustomTkinter* to provide an interactive graphical interface.

6. RESULTS AND DISCUSSION

6.1 Model Performance

The performance of the proposed sign language recognition system was evaluated using three machine learning algorithms: *Support Vector Machine (SVM)*, *Random Forest (RF)*, and *K-Nearest Neighbors (KNN)*. The models were evaluated using four performance metrics: *Accuracy*, *Precision*, *Recall*, and *F1-Score*. These metrics help to measure the classification performance of the system across the predefined sign language gestures.

□ Accuracy

Measures the proportion of correctly predicted samples out of all samples.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100$$

□ Precision

Measures the proportion of correctly predicted positive samples out of all positive predictions.

$$Precision = \frac{TP}{TP+FP} \times 100$$

□ Recall

Measures the proportion of actual positive samples correctly identified by the model.

$$Recall = \frac{TP}{TP+FN} \times 100$$

□ F1-Score

The harmonic mean of precision and recall, balancing both metrics.

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The results show that *SVM* achieved the highest performance, with an accuracy of 96.7%, precision of 96.7%, recall of 96.7%, and F1-Score of 96.7%. Random Forest achieved 94.7% across the evaluated metrics, while KNN achieved 89.4% accuracy, 89.7% precision, 89.7% recall, and 89.4% F1-Score.

Table 1. Performance Comparison of Machine Learning Algorithms

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
SVM	96.7	96.7	96.7	96.7
Random Forest	94.7	94.7	94.7	94.7
KNN	89.4	89.7	89.7	89.4

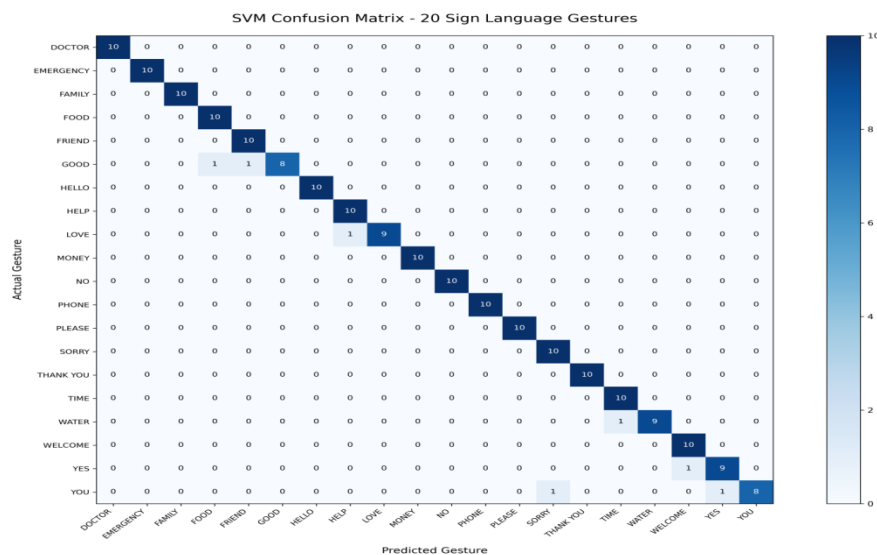
These results indicate that the three algorithms were able to classify the sign language gestures with different levels of performance. The obtained values are represented graphically in the corresponding *Accuracy, Precision, Recall, and F1-Score comparison graphs*.

6.2 Confusion Matrix Analysis

The confusion matrix was used to analyse the classification performance of the *SVM model* for the 20 predefined sign language gestures. The matrix contains *10 test samples for each gesture*, giving a total of *200 test samples*. The diagonal values represent correctly classified gestures, while the off-diagonal values represent misclassified gestures.

The SVM model correctly classified *192 out of 200 samples*, resulting in an overall accuracy of *96%* for this confusion-matrix test. Most gestures were correctly classified with a value of *10* on the diagonal. A few misclassifications were observed between *GOOD and FOOD, LOVE and HELP, WATER and TIME, YES and WELCOME*, and *YOU with SORRY and YES*.

The confusion matrix shows that the SVM model provides consistent recognition across most of the gesture classes. The small number of misclassified samples indicates that some gestures with similar hand positions or landmark patterns may be difficult to distinguish.

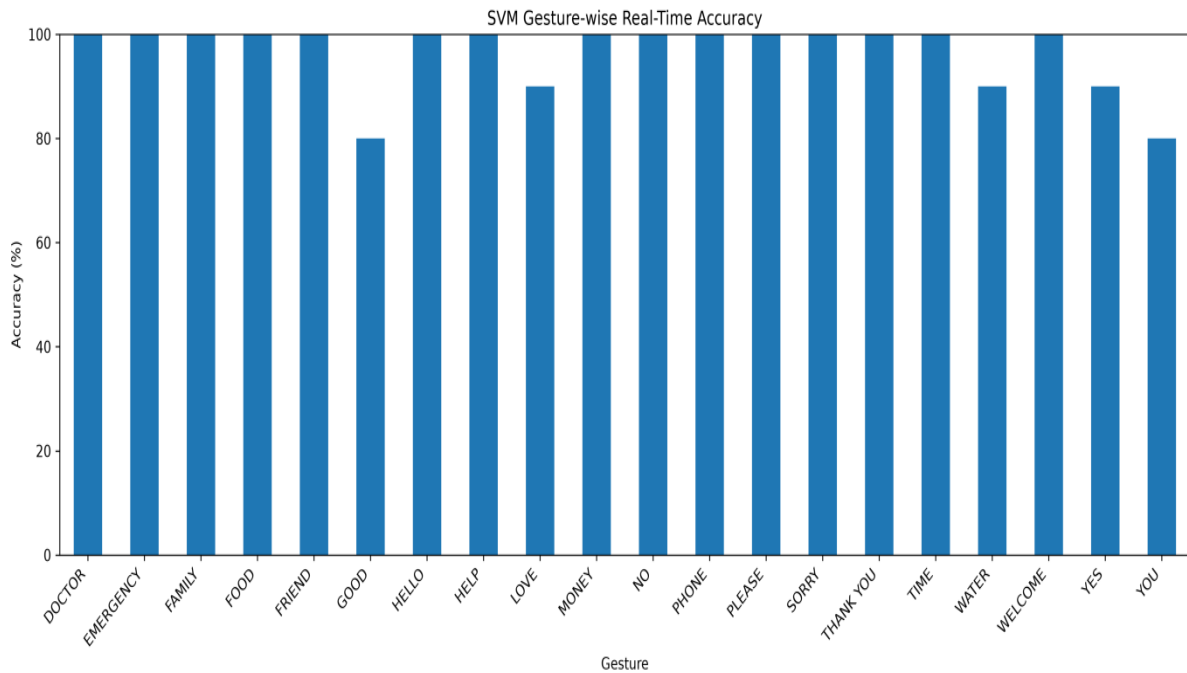
Figure 2 – SVM Confusion Matrix for 20 Sign Language Gestures

6.3 Gesture Recognition Results

The proposed system was evaluated for each of the 20 predefined sign language gestures using real-time webcam testing. The gesture-wise accuracy was calculated based on the successful recognition of each gesture during the testing process. The results show that most of the gestures achieved *100% recognition accuracy*.

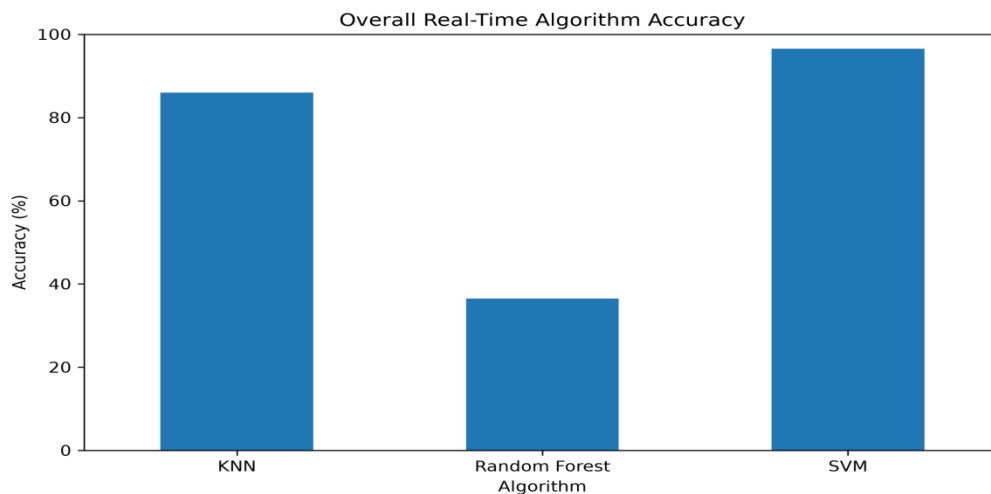
The gestures *DOCTOR, EMERGENCY, FAMILY, FOOD, FRIEND, HELLO, HELP, MONEY, NO, PHONE, PLEASE, SORRY, THANK YOU, TIME, and WELCOME* achieved 100% accuracy. The *GOOD* gesture achieved 80% accuracy, while *LOVE* and *WATER* achieved 90% accuracy. The *YES* gesture also achieved 90% accuracy, whereas *YOU* achieved 80% accuracy.

The results indicate that the system provides consistent recognition for most of the predefined gestures. The lower accuracy observed for *GOOD* and *YOU* suggests that these gestures may require more careful hand positioning during real-time recognition.

Figure 3 – SVM Gesture-Wise Real-Time Accuracy

6.4 Real-Time Performance

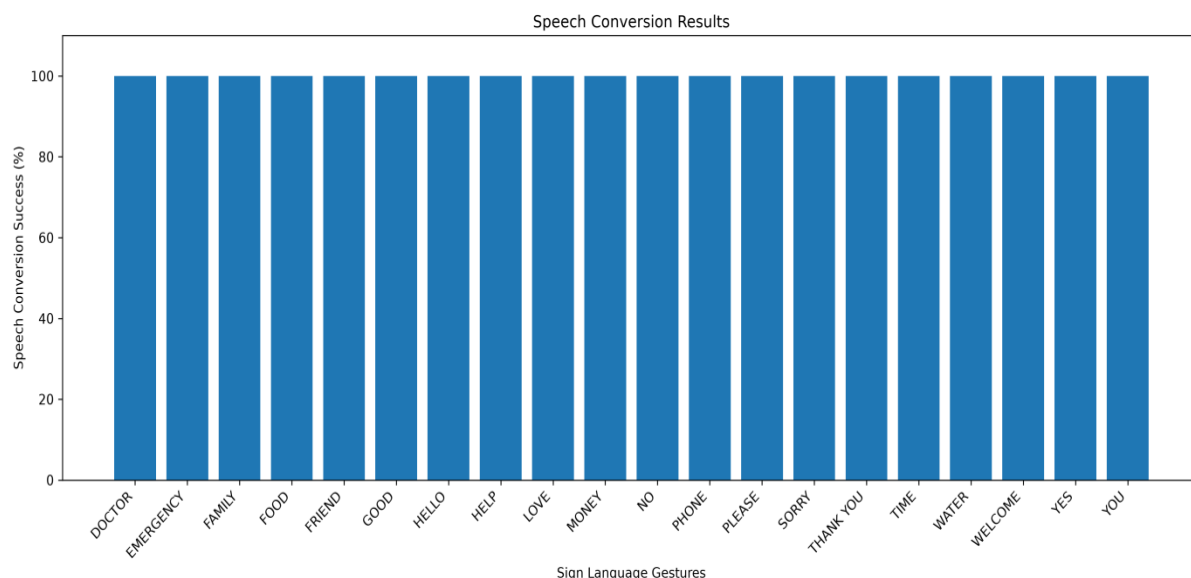
The system was tested using real-time webcam recognition. SVM achieved 96.5% accuracy, followed by KNN at 86.0% and Random Forest at 36.5%. The recognized gestures were displayed as text in real time. *The results demonstrate the practical performance of the proposed system during live gesture recognition.*

Figure 4 – Overall Real-Time Algorithm Accuracy

6.5 Speech Conversion Results

The speech conversion module was evaluated for the 20 predefined sign language gestures. Each recognized gesture was converted into spoken output using the text-to-speech component. The results show a *100% speech conversion success rate for all 20 gestures*. This indicates that the recognized gesture words were successfully passed to the speech conversion module and processed as audio output.

The evaluated gestures include DOCTOR, EMERGENCY, FAMILY, FOOD, FRIEND, GOOD, HELLO, HELP, LOVE, MONEY, NO, PHONE, PLEASE, SORRY, THANK YOU, TIME, WATER, WELCOME, YES, and YOU. The consistent 100% result demonstrates successful speech generation for all predefined gesture outputs.

Figure 5 – Speech Conversion Success Rate for the Predefined Sign Language Gestures

7. APPLICATIONS

The proposed sign language recognition system can be used in different areas where communication between sign language users and non-sign language users is required. It can be used in educational institutions to support communication between students, teachers, and staff. The system can also be useful in hospitals and healthcare environments for communicating basic requirements through recognized gestures. In public service areas, the system can assist users by converting predefined sign gestures into text and speech. It can also be applied in human-computer interaction systems, accessibility applications, and communication assistance tools.

8. BENEFITS

The proposed system provides several benefits for real-time sign language communication. It provides both text and speech output for the recognized gestures, making the system easier to understand. The use of MediaPipe landmarks reduces the need to process complete images during classification and provides numerical hand features for machine learning. The system supports both left and right hand inputs and can recognize 20 predefined static word-level gestures. The use of machine learning algorithms allows different classification models to be evaluated and compared. The real-time webcam-based operation also provides immediate recognition and feedback to the user.

9. LIMITATIONS

The proposed system has some limitations. It is designed to recognize only 20 predefined static word-level gestures and therefore cannot recognize all sign language vocabulary. The recognition performance may be affected by variations in hand position, orientation, distance from the camera, and lighting conditions. Similar-looking gestures may sometimes be difficult for the classifier to distinguish. The system requires a webcam for real-time operation. In addition, the current system focuses on static gestures and does not support continuous dynamic sign language sentences or complete sign language grammar.

10. FUTURE ENHANCEMENTS

The system can be enhanced by increasing the number of gesture classes and expanding the dataset with more samples and different users. Dynamic gesture recognition can be added to recognize gestures involving hand movement over time. Deep learning models such as CNN, LSTM, or Transformer-based architectures can be explored for more complex sign language recognition tasks. The system can also be extended to recognize facial expressions and body movements along with hand gestures. Support for multiple sign languages and multilingual text-to-speech conversion can be added. A mobile or web-based version can also be developed to make the system more accessible in practical applications.

11. CONCLUSION

The proposed work presents a machine learning-based real-time sign language recognition system with speech conversion. The system uses a webcam to capture hand gestures, MediaPipe to extract hand landmarks, and machine learning algorithms to classify the gestures. A dataset containing 4,000

samples of 20 predefined static word-level gestures was used for evaluation. KNN, SVM, and Random Forest algorithms were implemented and evaluated using standard performance metrics. The experimental results demonstrate that the system can recognize predefined gestures in real time and provide both text and speech output. The speech conversion module successfully converts the recognized gesture words into spoken output. Overall, the system provides a practical approach for assisting communication through real-time sign language recognition.

REFERENCES

The following references are supported by the research papers you uploaded and are suitable for this journal.

- [1] D. KhajehPourian and A. D. Zarei, "Sign Language Recognition using Modified Deep Learning Network: CNNs-LSTM," Scientific Reports, vol. 14, 2024.
- [2] H. Hu et al., "SignBERT: Pre-Training of Hand-Model-Aware Representation for Sign Language Recognition," arXiv preprint arXiv:2110.05382, 2021.
- [3] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, "Machine Learning Methods for Sign Language Recognition: A Critical Review and Analysis," Intelligent Systems with Applications, vol. 12, 2021, Art. no. 200056.
- [4] K. Bantupalli and Y. Xie, "American Sign Language Recognition using Deep Learning and Computer Vision," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 4896–4899.
- [5] K. Shenoy, T. Dastane, V. Rao, and D. Vyavaharkar, "Real-time Indian Sign Language (ISL) Recognition," arXiv preprint arXiv:2108.10970, 2021.
- [6] N. Gopinath, J. Anuja, S. Anusha, and V. Monisha, "A Survey on Hand Gesture Recognition Using Machine Learning," International Research Journal of Engineering and Technology (IRJET), 2020.
- [7] N. Rajasekhar, M. G. Yadav, C. Vedantam, K. Pellakuru, and C. Navapete, "Sign Language Recognition using Machine Learning Algorithm," 2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS), Coimbatore, India, 2023, pp. 303–306.
- [8] O. Vedak, P. Zavre, A. Todkar, and M. Patil, "Sign Language Interpreter using Image Processing and Machine Learning," International Research Journal of Engineering and Technology (IRJET), 2019.
- [9] P. Navendu and V. Sahula, "Word Level Sign Language Recognition using MediaPipe and LSTM-GRU Network," TechRxiv, July 2024.
- [10] V. Sarma Ch, H. Bindu Gogineni, B. Prasad, and P. Praveen Kumar, "Real-Time Recognition of American Sign Language Using Mediapipe and Machine Learning Techniques," Journal of Information Systems Engineering and Management, vol. 10, no. 3, 2025, pp. 1–15.