2025 2nd International Conference on Computational Intelligence, Communication Technology and Networking (CICTN) | 979-8-3315-3038-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/CICTN64563.2025.10932602

Real-Time Sign Language Recognition and Translation Using MediaPipe and Random Forests for Inclusive Communication

Garvit Sharma
Department of Computer Science and Enggineering, Graphic Era (Deemed to be University), Dehradun, India
Sharmagarvit372@gmail.com

Prakshep Gusain
Department of Computer Science and Enggineering, Graphic Era (Deemed to be University), Dehradun, India
prakshep.pg20@gmail.com

Aman Verma
Department of Computer Science and Enggineering, Graphic Era (Deemed to be University), Dehradun, India
aman.verma.smaa95@gmail.com

Harsha Saini
Department of Computer Science and Enggineering, Graphic Era (Deemed to be University), Dehradun, India
harshasaini474@gmail.com

Rishi Kumar
Department of Computer Science and Enggineering, Graphic Era (Deemed to be University), Dehradun, India
rishikumar.cse@geu.ac.in
https://orcid.org/0000-0002-1451-2686

Guru Prasad M S
Department of Computer Science and Enggineering, Graphic Era (Deemed to be University), Dehradun, India
guruprasad.cse@geu.ac.in

***Abstract*— Communication barriers faced by deaf and speech-impaired individuals necessitate innovative solutions for effective interaction with the hearing community. This research proposes a novel approach to sign language recognition and translation by leveraging Random Forests and MediaPipe technology. MediaPipe facilitates real-time hand gesture tracking and extraction of hand landmarks crucial for sign language interpretation. Subsequently, a Random Forest classifier is employed to recognize the signs depicted by the tracked gestures accurately. Upon successful recognition, the system seamlessly translates the sign language into written words and text, fostering enhanced communication accessibility for the deaf and speech-impaired community. Integrating advanced machine learning techniques with real-time tracking technology offers promising avenues for breaking down communication barriers and promoting inclusivity.**

Keywords*— *Sign Language; Recognition; Translation; Random Forest; MediaPipe; hand landmarks.

I. INTRODUCTION

Language is essential for deaf/hard-of-hearing people to communicate with the hearing world. Failure to understand sign language interferes with daily interactions, education, and social activities. To solve this problem, the search for sign language and machine translation emerged. They use technology to instantly recognize characters and translate them into other languages, such as speech or text. This system aims to improve communication, participation and access to information for deaf people. We have developed a system to detect and interpret signatures using Random Forests and MediaPipe. Google's MediaPipe helps us determine the hand landmarks. We use random forests to determine areas by evaluating cell values in the dataset. Once the character is recognized, the system converts it into text, making communicating more accessible for people. This research has three primary objectives:

1. Instant Sign Language Detection: Create a system that quickly and accurately recognizes the correct language description, ensuring effective communication in real-time.

2. Machine learning: Using random forest classification tools to identify unique characters by analysing features. The goal is to achieve accuracy in character identification.

3. Bridging the communication gap: Using appropriate techniques to translate visual symbols into text. This step ensures improved communication by making the language accessible to a broader audience.

The objectives mentioned above are designed to improve the language and translation skills among unique humans..

II. PROBLEM STATEMENT

Pinpointing the exact problem with any project is always a puzzle. It can be disclosed initially but understood as soon as it is processed. There are different ways of gathering information and defining problems, but the best way is to dig deeper and ask questions. This enables one to identify where the problem originates to apply the appropriate solution. Therefore, enough time and energy should be spent on problem identification in project management processes. Nowadays, sign language is used by many people who have hearing problems as their primary form of communication. This still does not fully close the gap between them and those from broader society. Such division presents many challenges for this population, from feeling lonely to being unable to access similar information and services because it is challenging to communicate with deaf and hard-of-hearing people.

Consequently, the deaf community has very few opportunities at large. Nevertheless, solutions available to bridge this communication abyss currently rely heavily on human interpreters, which are expensive, impracticable, or even not always accessible, whilst automated sign language recognition and translation systems face several problems. However, the existing systems often have challenges in accuracy that are attributed to complex handling of scenarios, especially when there are differences in hand shapes, orientations, lighting conditions and individual styles used.

1. Limited vocabulary support: Many of these systems only support a small number of signs, so they can only be effective within limited contexts.

2. Real-time translation difficulties: Most present sign language translation systems still find it hard to accurately recognize and translate sign language in real-time.

These limitations emphasize the need for a more robust and all-inclusive system for recognizing and translating sign languages. The proposed system should mainly address the following:

1. Achieve high accuracy in sign recognition: It involves accurately recognizing individual signs under different environmental settings and signing styles.

2. Enable real-time translation: Detected signs should be accurately translated into correct characters with minimal delay, promoting non-stop communication at work.

3. Be user-friendly and accessible: The system should be designed to be easy to use, making it readily available and accessible to users with varying technical expertise and abilities.

Meeting these challenges and developing a robust and comprehensive sign language detection and translation system can significantly improve the lives of individuals with hearing impairments, fostering greater inclusivity and equality. of the current designations.

III. LITERATURE SURVEY

The hand and gesture recognition journey began as early as 1987 when Zimmerman, Lanier, Blanchard, Bryson, and Harvill proposed an interface that could estimate a hand's position and orientation using a glove's magnetic sensors. Fast forward to recent years, the emergence of deep learning has sparked a surge of research aimed at improving hand sign language recognition. This has been evidenced by the works of various researchers including Elboushaki, Hannane, Afdel & Koutti (2020), Gomez-Donoso, Orts-Escolano & Cazorla (2019), Kopuklu and colleagues (2019), Lim, Tan & Tan (2016), Rastgoo, Kiani & Escalera (2018), and Yuxiao, Zhao, Peng, Yuan & Metaxas (2019) [1].

People use different methods to recognize language. One way is Hidden Markov Models (HMMs). They have variations like multi-stream HMMs (MSHMMs), Light-HMMs, and Bound Mixture Density HMMs. Other models include neural networks, artificial neural networks (ANN), Naive Bayes classifiers (NBC), and multilayer perceptrons (MLP). There are also Self-organizing maps (SOM), self-organizing feature maps (SOFM), and simple recurrent networks (SRN). Support vector machines (SVMs) and 3D convolutional residual networks are essential, too. Apart from these, researchers also tried the wavelet method and eigenvalue Euclidean distance [2].

Different processes or machines produce different levels of accuracy. For example, the accuracy of Light-HMM reaches 83.6%, MSHMM reaches 86.7%, SVM reaches 97.5%, the eigenvalue reaches 97%, and the wavelet family tops the list with 100% [2]. While these models show high accuracy, it is essential to note that accuracy does not depend on the model used. It is also affected by factors such as the size of the dataset and the clarity of the dataset image (these factors may vary depending on the method, tools, etc. of the data taken). While the numbers are encouraging, there are all the details of the game behind the scenes to hit the right notes [2].

Speech recognition is drawn from deep learning techniques. Since YOLO is easy to train, it's a deficiency on the front that the model is regarded as poor in accuracy and uses unsuitable training. Therefore, this YOLO mode is an effective and widely used one-step product monitoring and analysis method. YOLOv5 is the latest version of the YOLO architecture that executes training safely and securely and consumes less time and resources. The YOLO models are the most commonly used in language training studies. Indeed, Dima and Ahmed (2021) brought the YOLOv5 algorithm into play to build an image processing pipeline for ASL recognition that has resulted in satisfactory recognition rates with a mean accuracy of 96.6% over 500 images. In another research, the YOLOv3 algorithm was employed to measure the MSL (Malaysian Sign Language) level, and this approach was proven by 93% at 300 images level [3].

A.K. Sahoo and his fellow researchers emphasized machine learning to encounter Indian Sign Language (ISL). For the study, these scientists looked into static hand movements depicting the numbers 0 up to 9. RGB sensors were used to acquire images of these characters, and more than five hundred images were collected. Each image was assigned a number, which was the basis for the game. To learn the model, the team deployed supervised learning techniques of Naive Bayes and the nearest neighbour. The average accuracy for the models is 98.36% for the k-nearest model and 97.79% for the Naïve Bayes model, which gives the k-nearest a slight edge over Naïve Bayes [4].

Sundar B and his crew proposed a vision-oriented solution to ASL text comprehension with the help of the Media Pipe library. They used a gesture recognition model based on long-term memory (LSTM) to identify 26 letters of ASL and achieved 99% accuracy (99%). However, transferring the plan of action directly into the text format makes it perfect for human-computer interaction (HCI). Integration of the gestures of Media Pipe and LSTMs has shown to be very helpful concerning navigation recognition in HCI applications [5].

In another experiment, Jyotishman Bora and his group classified Assamese sign language using mechanized learning. Combinations of 2D and 3D videos and MediaPipe's method of hands-on solution were used for training neural networks. This model acquires a 99% accuracy in recognizing Assamese gestures. This research discloses that their method approach is beneficial even with other texts and gestures, and it can also be used for other languages. MediaPipe solutions ensure accurate real-time monitoring of body movement and help quicker deployment, thus allowing the use of rugged platforms while keeping the speed and accuracy rate [6].

Using the 3D CNN architecture and RGB videos, Huang and the team proposed to generate sentences from a hierarchical network (HAN) and hidden space learning with both physical and rhyming words. While they did not enhance the data by image conversion, it would have increased the accuracy of the 82.7% they achieved. Cruz, however, achieved 83.75% accuracy in large scale English (BSL) identification in over 23,000 images and 560 characters using 3D contrast using a CNN with adaptive learning and data augmentation. Although the mentioned alterations may be considered a good beginning, there is a place for more research, such as changing backgrounds and testing the character development technique, to have the best effect on recognition. Mainly, previous reports do not explain in detail how the data collection and modelling was done; maybe it can be helpful for future research that involves massive data sets and symbols translation [7].

In 2019, Sruthi and Lijiya embarked on a thrilling journey to establish a technology capable of recognizing word-for-word text in Indian Sign Language (ISL). Note that hands or fingers do not move these letters. They had a set of 24 symbols to address, and they used their deep learning capability to gain 98.64% accuracy. Scientifically gathering image datasets for model training is quite intriguing. To begin with, they undertake face detection to locate faces in the image and reduce the pixels around the faces by blurring them. After that, they carried out the cell region segmentation with the colour segmentation algorithm and also the maximum connectivity algorithm. After making these changes, the dataset can be described to their model, which they call the "signature model," where it is trained to recognize different locations in the dataset. This blows my mind on how they utilize technology to fill the communication void [8].

IV. METHODOLOGY

Data pre-processing and feature extraction: The starting point of the research is to create a database of hand gestures that will be used to train machine learning models. The capture.py file is the key that holds the file collection process together. This Python code relies on the OpenCV library to load and retrieve images from the webcam, allowing us to engage the user.

The script iterates through several predefined classes, which are the sign representations on which the system is trained. The code displays visual prompts on the screen to notify the user when it is ready to ensure that the sign communication does not fail due to insufficient user preparation. Then, the people in charge execute the script, capturing 100 images in each sign class. These snapshots are organized and placed in a directory. Each category has a folder created specifically for that category for ease of storage and access. We have used the MediaPipe framework for predictive modelling and converting lists of images into a format suitable for machine learning models.

Data iteration: The first program cycles through all image files in the directory, representing hand gestures.

Hand Landmark detection: This uses MediaPipe, a real-time hand sign detection library widely used in many applications. MediaPipe does an excellent job of determining the area of the hand because it covers the positions of twenty-one critical landmarks on the hand; most of these are the fingertips, middle of the palm and wrist.

Coordinate extraction: The code stores the x and y coordinates of the keys for each hand. These coordinates show the exact location of the landmarks in the image.

Data normalization: Through the standardization process, we can address the issue of image size and hand location variance. The code obtains the x and y values by subtracting the minimum value found across all the image landmarks. Therefore, this process converts the hand data points into a relative coordinate system with spatial relationships among the landmarks rather than their absolute location in the image frame.

Data persistence: Lastly, the data of processed landmarks clubbed together with the sign class label (which mainly indicates the sign captured in the image). After that, the data is combined and saved into a picked file. The pickling format improves the efficiency of storing and getting access to the processed data, making it available for future use. This data type could function as a training unit using a machine learning model like a Random Forest classifier that could detect the hand pose features extracted from the images.

Gesture Classification: This subsection provides a procedure for training and testing a Random Forest classifier for sign language categorization. The code uses previously processed landmark information from the hand and corresponding signs. These labels and information are obtained from the pickle file created during the Data processing stage. The processed data involves 2D coordinates of the critical points on the hand, representing the relative position of these key points during sign gestures. This code uses a feature selection/dimensionality reduction step to maximise training within the Random Forest model. It transforms 2D spatial data (arrays for x and y coordinates) into a single, one-dimensional array. It can be stated that this flattened array represents a feature vector, which in turn captures the hand pose info for every sign model. After that, the code divides the processed data into training and testing segments. This is a significant step that helps to train the model on a sample of the data. The evaluation of unseen datasets would assess the model's ability to generalize. Stratifying is a technique that checks the class balance across training and testing sets. It is most likely used during the dividing phase. This promotes training the model on a balanced dataset without a bias towards more common sign occurrences. Finally, a Random Forest classifier is fitted on the specified training data set. This model takes advantage of decision trees' power to learn these underlying features in hand poses and conclude on the corresponding sign. Next, the trained model is tested using data from the test set. Lastly, the trained model is saved and made accessible for future usage.

Detection and Translation: In the last section, we go through the development of a real-time American Sign Language (ASL) recognition system. The system utilizes a webcam video as input and attempts to convert ASL gestures into text in real time as output.

To accomplish results, the code builds a multi-stage processing pipeline. The first stage revolves around video frame processing. Each captured frame from the webcam or video is subjected to hand detection and tracking within the image. This can be achieved by applying computer vision techniques that use algorithms for skin colour segmentation or hand shape analysis. When hands are finally discovered, the program extracts vital features from them. This is the task of finding and marking out particular points on the hand, sometimes called hand landmarks. MediaPipe, the mighty real-time hand pose estimation framework, is used to fulfil this task. By taking these landmarks (like knuckles, palm centre, and wrist), the system delivers fundamental data about the hand pose in the context of sign language gestures. Following the hand landmark extraction process, the system employs pre-trained machine learning algorithms which study the extracted features and identify the corresponding letters or numbers. Here, the trained Random Forest classifier takes up a central position. By comparing the extracted hand pose features with the patterns learned during training, the model decides on the most probable performed sign. In the end, the predicted symbols are converted to letters. This step converts the recognized ASL gestures (letters or numbers) to a humanreadable format and thus fills in the communication gap between sign language and spoken language. This real-time functionality enables smooth conversations and promotes inclusivity in deaf and hard-of-hearing individuals.

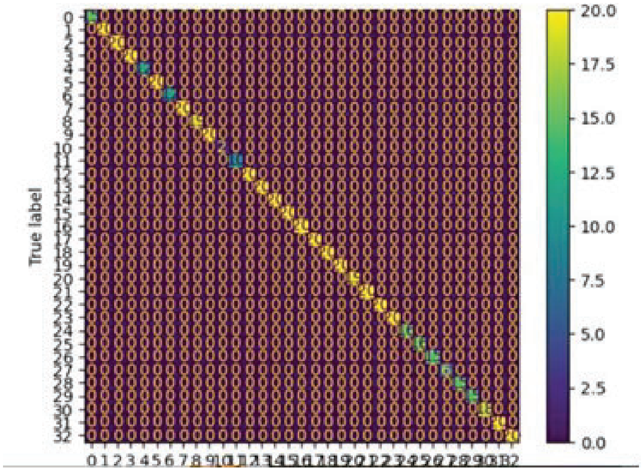
V. RESULT AND DISCUSSION

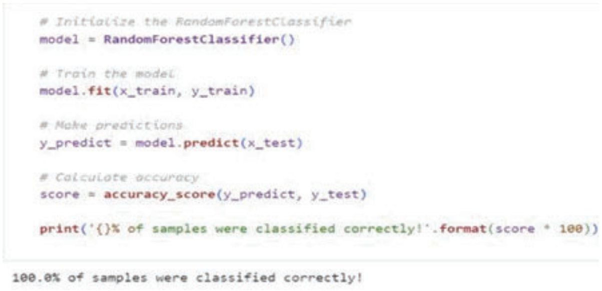
Dataset: The dataset used to train the model was taken from multiple online sources containing images under different lighting conditions for different hand shapes and complexions. The dataset was divided into 33 classes containing images for 26 alphabets and 7 numbers. Each class of images comprised around 100 images per sign for an alphabet or a number. The experiment evaluated the performance of several machine learning algorithms for sign classification alongside Mediapipe for landmark extraction, which included Random Forest, Decision Tree, Logistic Regression, KNN, and SVC. The results obtained on the test dataset are summarized below:

TABLE I. CONFUSION MATRIX OBTAINED FOR THE RANDOM FOREST CLASSIFIER MODEL ON THE TEST DATA SET

Algorithm	Accuracy
Random Forest	100%
Decision Tree	99.4736%
Logistic Regression	94.9122%
SVC	98.2456%
KNN	99.8245%

The confusion matrix obtained for our custom-made dataset using the Random forest Classifier model alongside MediaPipe for landmark extraction for 33 different classes of signs is as follows:





```
# Initialize the RandomForestClassifier
model = RandomForestClassifier()

# Train the model
model.fit(x_train, y_train)

# Make predictions
y_predict = model.predict(x_test)

# Calculate accuracy
score = accuracy_score(y_predict, y_test)

print('{}% of samples were classified correctly!'.format(score * 100))

100.0% of samples were classified correctly!
```

Fig. 1. Accuracy obtained for Random Forest

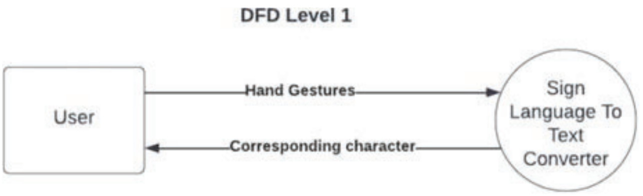


Fig. 2. Data Flow Diagram

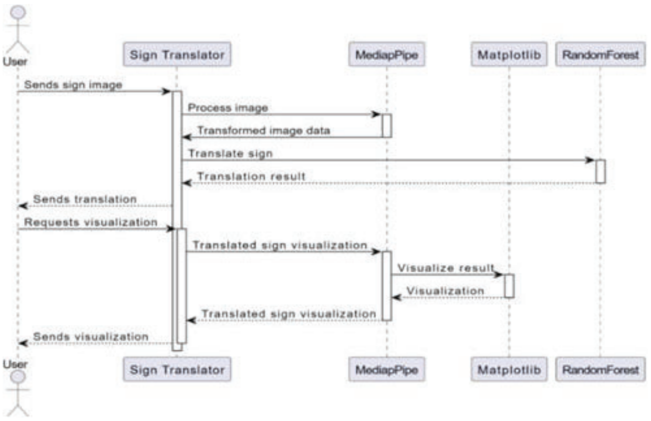


Fig. 3. User Interaction with Sign Translator

This Fig.3 illustrates the sequence of interactions in a sign language translation system. The user sends a sign image to the Sign Translator, which processes it using MediaPipe for image transformation and sign translation, aided by a Random Forest model. The translated result is sent back to the user, and if visualization is requested, Matplotlib generates and returns a visual representation of the translation.

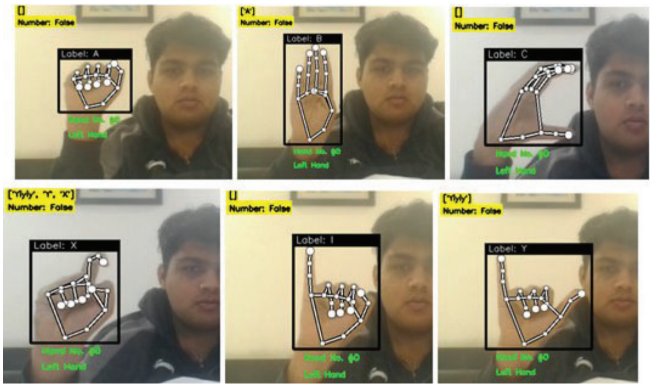


Fig. 4. Sample working with User

Fig.4 displays live sign translator system that recognizes and interprets hand gestures corresponding to different sign language letters (A, B, C, X, I, and Y) using a hand tracking framework. Each frame shows the hand's skeletal structure, labeled with its respective letter, alongside real-time details such as hand orientation and position (e.g., left hand). The system used for gesture-based communication or educational purposes.

VI. CONCLUSION

As a result, sign interpreters developed using standards such as Mediaipe, OpenCV, and random forest proved effective for a small dataset comprising just 100 images per sign. The system successfully addresses the issue of varying lighting conditions and different hand shapes. This effect can be credited to using the mediapipe library, which is remarkable at detecting the 21 hand landmarks even in low lighting conditions. The research analyzed the performance of machine learning classifiers like Random forest classifier, Decision trees, Logistic Regression, SVC and KNN alongside the Mediapipe framework. It was concluded that a bagging algorithm like random forest performs the best in terms of accuracy obtained for the used dataset. In the future, neural networks like LSTM and RNN can be used to enhance this project and address the requirement of temporal dependencies. As this work continues, we look forward to a future where translators will become essential tools for aiding communication and fostering a connected and compassionate society.

REFERENCES

[1] R. Rastgoo, K. Kiani, S. Escalera “Hand sign language recognition using multi-view hand skeleton” Expert Syst. Appl., 150 (2020), Article 113336

[2] Anand and N. P. Singh, "Sign Language Recognition System using TensorFlow Object Detection API," International Journal of Engineering and Advanced Technology (IJEAT), vol. 9, no. 6, pp. 2009-2016, 2020.

[3] ADDSL: Hand Gesture Detection and Sign Language Recognition on Annotated Danish Sign Language Sanyam Jain Østfold University College Halden Norway 1783 sanyamj@hiof.no

[4] A. K. Sahoo, “Indian Sign Language Recognition Using Machine Learning Techniques,” Macromolecular Symposia, vol. 397, no. 1. Wiley, p. 2000241, Jun. 2021. doi: 10.1002/masy.202000241.

[5] Sundar, B., & Bagyammal, T. (2022). American Sign Language Recognition for Alphabets Using MediaPipe and LSTM. Procedia Computer Science, 215, 642–651. https://doi.org/10.1016/j.procs.2022.12.066

[6] Bora, J., Dehingia, S., Boruah, A., Chetia, A. A., & Gogoi, D. (2023). Real-time Assamese Sign Language Recognition using MediaPipe and Deep Learning. Procedia Computer Science, 218, 1384–1393. https://doi.org/10.1016/j.procs.2023.01.117c

[7] Efficient sign language recognition system and dataset creation method based on deep learning and image processing A. L. Cavalcante Carneiroa , L. Brito Silvaa , D. H. Pinheiro Salvadeoa aDept. of statistics, applied mathematics, and computation, State Univ. of São Paulo, Av. 24 A, 1515, Rio Claro, SP, Brazil, 13506-700

[8] S. Das, M.S. Imtiaz, N.H. Neom, N. Siddique, H. Wang, a hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier, Expert Syst. Appl., 213 (2023), Article 118914

[9] “Treatment episode data set: discharges (TEDS-D): concatenated, 2006 to 2009.” U.S. Department of Health and Human Services, Substance Abuse and Mental Health Services Administration, Office of Applied Studies, August, 2013, DOI:10.3886/ICPSR30122.v2