


A Novel Approach for Recognizing Real-Time American Sign Language (ASL) Using the Hand Landmark Distance and Machine Learning Algorithms

Shahan Ahmed 

*Dept. of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
shahan.ahmed001@gmail.com
<https://orcid.org/0009-0006-9593-2119>*

Sumit Kumar Kar 

*Dept. of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh
sumitkumarkar01@gmail.com
<https://orcid.org/0009-0000-9469-3108>*

Sarnali Basak

*Dept. of Computer Science and Engineering
Jahangirnagar University
Savar, Bangladesh
sarnali.cse@juniv.edu*

Abstract—Several sign language recognition techniques and models have been prevalent to ease the communication gap between people with hearing disabilities and people who don't understand sign language. The proposed model is a vision-based approach aims to find a better system to recognize sign gestures using the distance between hand landmarks. This process doesn't rely on complex image processing techniques, making it robust towards challenging lighting conditions, noisy backgrounds, and image resolution. To accurately detect 26 American Sign Language (ASL) letters, the model only extracts 12 features. As a result, it can recognize the letters more rapidly than the traditional image-based method and can be trained with lesser computational resources. Therefore, low-configured devices can likewise utilize the method described in this paper. Despite a number of sign language recognition methods are currently in widespread use, the inability of those models to handle the orientation of some specific letter combinations (D, G, U, H, I, and J) has a severe effect on their overall effectiveness. This proposed algorithm, added with features 11 and 12, can solve the anomalies in the orientation of similar pair alphabets. Therefore, the four most popular algorithms, Naive Bayes, Support Vector Machine (SVM), Random Forest, and K-Nearest Neighbors (KNN), have been used to predict the letters of the American Sign Language alphabet. Regardless of the algorithm employed, the model was highly efficient because of the carefully curated dataset, which included attributes directly related to the ASL alphabet signs. SVM demonstrated the best accuracy among these algorithms, outperforming the others by 97%.

Keywords—Hand Landmark, Machine Learning, MediaPipe, Sign Language Recognition, Vision Based, Hand Gesture, American Sign Language (ASL).

I. INTRODUCTION

People with hearing disabilities struggle to communicate due to their inability to use spoken language. Among all other

communication methods for dumb and deaf people, sign language is one of the most popular communication mediums, as it can also express emotions. An estimated 466 million people among them 34 million are children, who have debilitating hearing loss worldwide, which is more than 5% of the world's population. And what's even more concerning is that this number will be projected to increase to over 700 million by 2050 [1]. Deaf people utilize various communication methods depending on their preferences, cultural background, and level of hearing loss. The most popular approaches include sign language, lip reading, tactile techniques, and written text. In sign language, gestures or symbols are arranged linguistically. A sign is any gestural communication. The three components of any sign are hand shape, hand placement, and hand movement [2] [3]. No universal sign language exists though more than 300 are used worldwide [4]. The two most researched methods of sign language recognition are sensor-based [5] and vision-based [6]. The issue with the sensor-based technique is that signers must wear sensor-attached gloves, which may only be appropriate in some contexts. On the other hand, the biggest challenge for the vision-based approach is that it has to recognize hands against the background. Many algorithms for sign language involve sophisticated image-processing methods that demand a lot of computational resources. In this research, the ASL alphabet is recognized in real-time from a fixed frame-rate video utilizing much lower computational resources. The methodology in this study combines OpenCV and MediaPipe to identify hand landmarks to extract 12 features using the methods described in Feature Extraction, and finally, it employs a variety of Machine Learning (ML) algorithms to identify ASL alphabet signs.

II. LITERATURE REVIEW

Several attempts have been made to recognize sign language, and many models have been proposed. The paper by Halder et al. [7] and Bora et al. [8] uses MediaPipe to detect palms and hand landmarks, followed by using a neural network technique, whereas Zhu et al. [9] use cross-resolution knowledge distillation method to achieve sign language recognition. Devdatta et al. [10] uses CNN-based methods to detect hand gestures, extract features, and classify the signs, and uses NLP techniques to figure out the grammar and structure of the sign language and convert it to text. Likewise, Athania et al. [11] applies SIFT algorithm for Feature Extraction using Softmax Classifier. A combination of CNN and LSTM models is applied by Bantupalli and Xie [12] to obtain the detection of sign characters. On the other hand, Rachana et al. [13], Hossein and Ejaz [14] work on grayscale images to achieve better classification by CNN. On the contrary, Muniraj et al. [15] also transformed the input image from RGB to YCbCr to improve skin identification, and finally, CNN was used to classify the images. Equivalently to differentiate between skin and background, Jiang and Ahmad [16] used HSV color, and then modeled by Support Vector Machine for classification.

III. METHODOLOGY

A. System Architecture

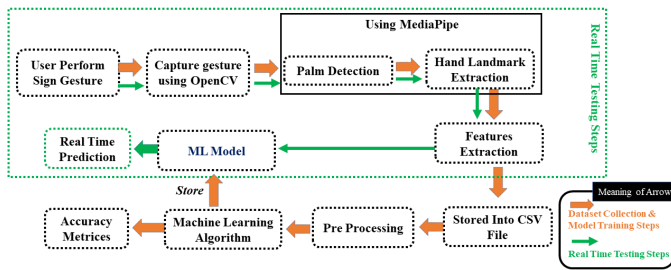


Fig. 1. Overview of Methodology

Fig.1 shows our approach to recognizing American Sign Language (ASL). A sign language performer performs various signs, which are then captured using OpenCV, followed by Mediapipe to detect palm and hand landmarks. Afterward, features are extracted using the process mentioned in Feature Extraction. Later, these features are fed to various ML algorithms in order to predict signs in real-time.

B. Feature Extraction

For capturing the gesture or sign, performed by a user who is dumb and deaf, OpenCV is used, and then MediaPipe library is used to detect and get the x and y coordinates of the 21 landmarks (LM), as shown in Fig. 2.

Initially, ten features are taken to identify the signs of our model. We use Table I and eq. (1) to calculate those ten features as shown in Fig. 3. Here, we use a ratio of Euclidean Distance (ED) between two specific Landmarks (LM) and Base Distance (BD) to calculate each feature from eq. (1).

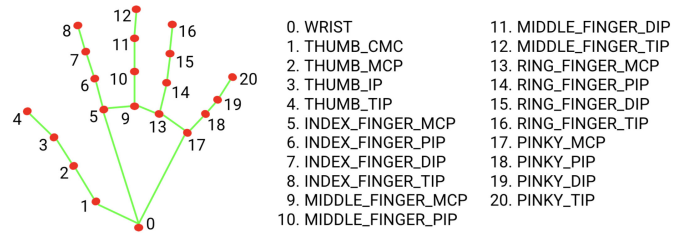


Fig. 2. MediaPipe Hand Landmarks

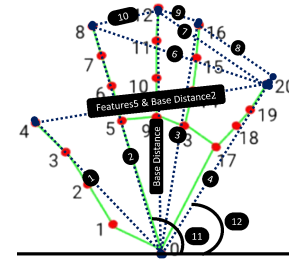


Fig. 3. Feature Extraction Points

$$Feature = \frac{ED \text{ between Point A and Point B}}{Base \text{ Distance}} \quad (1)$$

TABLE I
FEATURE TABLE

Feature	BD	Point A	Point B
1	BD1	LM0	LM4
2	BD1	LM0	LM8
3	BD1	LM0	LM16
4	BD1	LM0	LM20
5	BD1	LM4	LM20
6	BD2	LM8	LM20
7	BD2	LM12	LM20
8	BD2	LM16	LM20
9	BD2	LM16	LM12
10	BD2	LM18	LM12

Two Base Distances are calculated using Table II and Euclidean Distance (ED) eq. (2) between two landmarks. BD1 and BD2 are the longest ED between two LMs vertically and horizontally, respectively, so we use these two as base distances.

$$ED = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2)$$

TABLE II
BASE DISTANCE

Base Distance	Point A	Point B
BD1	LM0	LM12
BD2	LM4	LM20

Using these ten features, we could predict most of the signs in the English alphabet with spectacle accuracy. The

accuracy drops when some pair of signs are similar but has different orientations shown in Fig. 4, i.e. IJ, DG, and UH pairs. Therefore two more features (namely Feature 11 and Feature 12) assist in detecting those complex signs accurately.

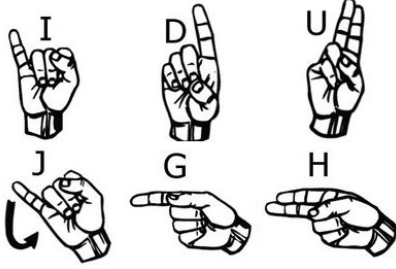


Fig. 4. ASL Sign Pairs with Same Hand Shape but Different Orientation

Algorithm 1: Algorithm for Feature 11 and Feature 12

Data: Performing Sign (PS), Index Finger (IF), Middle Finger (MF), Little Finger (LF)
Result: Feature 11, Feature 12
while ($PS = true$) **do**
 $Feature11 = 0$;
 $Feature12 = 0$;
 if (IF is up = true) or (MF is up = true) **then**
 if ($Angle1 \geq 0.8$) **then**
 $Feature11 = 1$
 else
 $Feature11 = 0$
 end
 if (LF is up = true) **then**
 if ($Angle2 \geq 0.8$) **then**
 $Feature12 = 1$
 else
 $Feature12 = 0$
 end
 end
end

Algorithm 1 is used to calculate Feature 11 and Feature 12. In order to figure out Angle 1 and Angle 2, a horizontal line across the frame at the lowest vertical position is drawn. Subsequently, the horizontal line is shifted to the y-coordinate position of LM 0 to create a Base Line (BL). Then two lines are drawn from LM 0 to LM 8, which is denoted as Line-1, and LM 0 to LM 20, which is denoted as Line-2, as shown in Fig. 5. Therefore, the angle between Line-1 and BL and between Line-2 and BL were calculated for Angle 1 and Angle 2, respectively, as shown in Fig 5. Due to the natural position of the ASL at the resting position, certain threshold values are enough to anticipate whether hands are rotated or not, regardless of the actual angle. The sample dataset is shown in TABLE III.

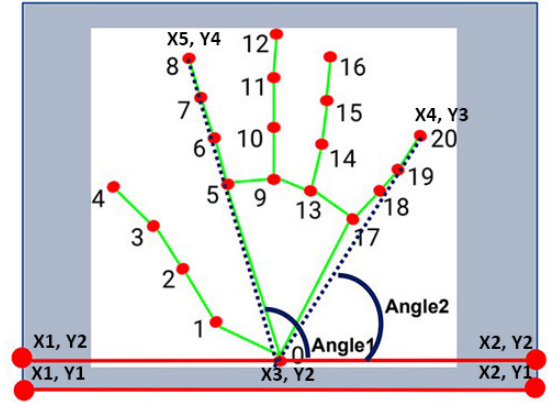


Fig. 5. Angle 1 and Angle 2 Calculation

C. Dataset Creation

In order to build the dataset, six male and two female participants performed the sign using hands. The entire hand gesture procedure of the participants is recorded and conducted by the MediaPipe while collecting the data. Hence the data is stored in a password-protected online storage to ensure data integrity and confidentiality. The dataset consists of 4121 data points related to ASL. We manually excluded some data points due to human error while collecting data. A single data point is comprised of 12 attributes and one target class. The dataset contains signs for all 26 English Alphabet. It is assembled using a high-resolution webcam (1280 x 720) recording at a frame rate of 30 fps. The ASL signers are instructed to perform each sign once, then the data collection process simultaneously collects 20 data points. Then, data is stored in a Comma Separated Values (CSV) file using the methods discussed in Feature Extraction. The dataset is divided into training and testing sets, with 80% for training and 20% for testing. Table III shows a portion of the dataset.

IV. RESULT ANALYSIS

A. Quantitative Analysis of the Model

The suggested ASL identification system is evaluated using important performance metrics (PM) such as accuracy, precision, recall, and F1 score under 10-fold cross-validation. This rigorous process ensures a thorough evaluation of the model's capabilities. The model is initially trained and tested on various datasets of individual signers to assess the system's competence. Following that, its overall efficacy is tested on a combined dataset containing gestures from all participants, allowing for a thorough analysis of the model's flexibility.

Table IV shows that Naive Bayes (NB) algorithm is the best overall performer for individual persons' dataset by getting an average from 8 distinct persons, achieving over 96% accuracy, precision, recall, and F1 scores. Random Forest (RF) algorithm is closely followed by NB, with over 96% performance across all metrics. KNN also performs admirably, achieving accuracy, precision, recall, and F1 scores of more than 95%. SVM

TABLE III
SAMPLE DATASET

SL No	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7	Feature 8	Feature 9	Feature 10	Feature 11	Feature 12	Alphabet
1	1.54	1.123	0.89	0.87	1.014	0.756	0.493	0.246	0.247	0.266	0	0	A
23	0.437	0.94	0.931	0.803	0.37	0.777	0.715	0.431	0.293	0.295	0	0	B
82	0.897	2.202	0.911	0.938	0.456	2.905	0.895	0.463	0.447	2.676	0	0	D
128	2.266	3.318	0.927	1.052	1.593	1.51	0.44	0.2	0.243	1.485	1	0	G
144	0.624	0.917	0.312	0.319	0.336	1.819	2.039	0.241	2.064	0.401	1	0	H
177	1.455	1.126	0.82	1.776	0.332	1.966	2.371	2.943	0.574	0.566	0	0	I
186	1.801	1.372	0.995	3.252	2.619	1.051	1.038	0.938	0.123	0.152	0	1	J
...
...
4000	1.534	1.399	0.851	0.929	0.928	0.977	0.527	0.224	0.329	0.491	0	0	T
4013	0.44	0.958	0.28	0.214	0.242	3.1	3.261	0.379	3.005	0.393	0	0	U
4121	1.39	2.811	0.626	0.516	0.959	2.416	0.622	0.23	0.409	1.927	0	0	Z

TABLE IV
PM FOR INDIVIDUAL PERSONS' DATASET

PM	NB	RF	KNN	SVM
Accuracy	96.15%	96.16%	95.78%	87.89%
Precision	96.98%	96.29%	95.60%	88.93%
Recall	96.15%	96.16%	95.78%	87.89%
F1	96.19%	96.15%	95.24%	87.60%
CV	96.71%	96.07%	95.57%	88.26%

exhibits relatively lower performance, achieving results over 87% for accuracy, precision, recall, and F1 scores.

TABLE V
PM FOR COMBINED DATASET

PM	NB	RF	KNN	SVM
Accuracy	80.34%	95.75%	96.72%	97.20%
Precision	81.47%	95.77%	96.88%	97.27%
Recall	80.34%	95.75%	96.72%	97.21%
F1	79.93%	95.74%	96.70%	97.21%
CV	81.17%	96.12%	96.17%	96.63%

Table V highlights the performance of various ML algorithms on the combined dataset. In this case, SVM is the best overall performer, with more than 97% accuracy, precision, recall, and F1 scores. SVM is closely followed by KNN, which reaches over 96% in all performance metrics, while RF achieves over 95% in all performance metrics. However, due to the combination of datasets, NB loses performance and reaches over 79% in all performance metrics. The ASL recognition model is first tested and evaluated on individual persons' datasets to measure its performance on a per-person basis. This technique gives insights into the ability of the model to capture and recognize ASL alphabet gestures for each individual accurately. The distinct datasets are gradually integrated into various complete datasets to assess further the model's effectiveness in a more realistic environment. These distinct combined datasets seek to simulate a greater diversity of signing styles and gestures to demonstrate how the model responds to the combination of various datasets. This combined dataset evaluation also provides insights into

the model's general robustness and adaptability, reflecting its performance when ASL recognition systems confront various users in real-life circumstances. Fig. 6 provides an overview

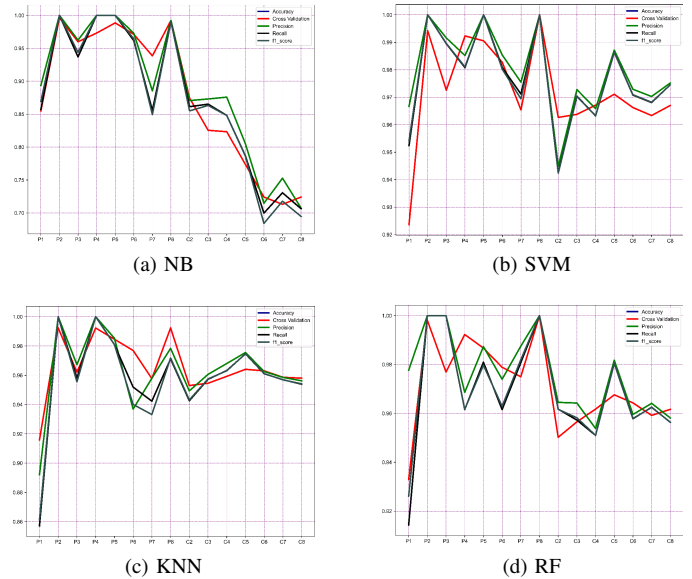


Fig. 6. Overall Performance for Individual and Combined Dataset

of the overall performance of the ML algorithms. On the X-axis, we have the individual datasets (P1 to P8) and gradually combined datasets of all participants (C2 to C8), while the Y-axis represents the corresponding percentage of performance metrics. The dataset is initially collected from eight individuals, and then various ML algorithms are applied to evaluate their performance for individual persons' datasets and combined datasets. Fig. 6 also illustrates the performance differences between the ML algorithms used on individual and combined datasets. Following the combination of the dataset, SVM outperforms KNN, RF and NB. NB initially works well for individual persons' datasets but, loses performance after aggregating datasets while maintaining decent performance metrics. These findings emphasize the need to evaluate the efficacy of ASL recognition algorithms utilizing individual and combined datasets.

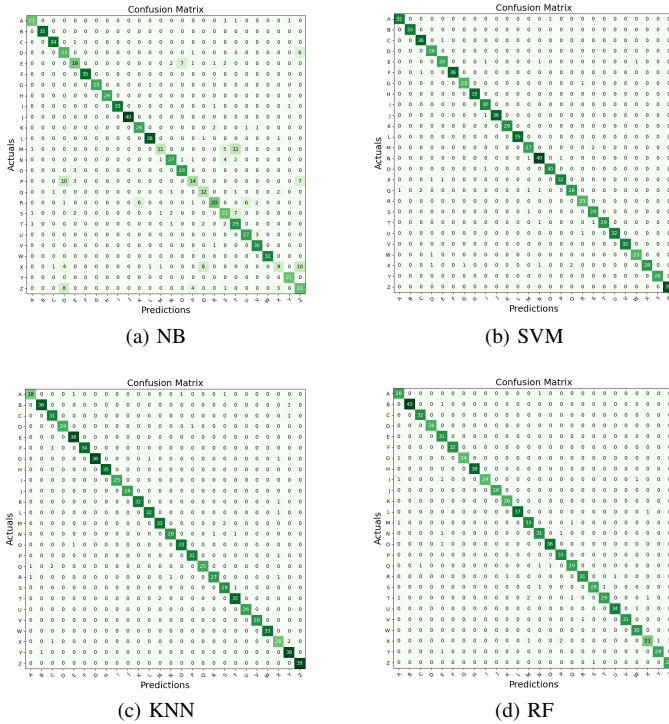


Fig. 7. Confusion Matrix for Combined Dataset

Fig. 7 depicts the confusion matrix, which provides insights into the relationship between the predicted and actual alphabets based on the combined information. It is worth noting that SVM performs notably, accurately recognizing all alphabets, including varied orientations. KNN and RF follow closely behind SVM, predicting accuracy. While NB provides generally promising performance, there are times when it struggles to distinguish certain complex signs, resulting in occasional confusion. Fig. 8 shows the Receiver Operating

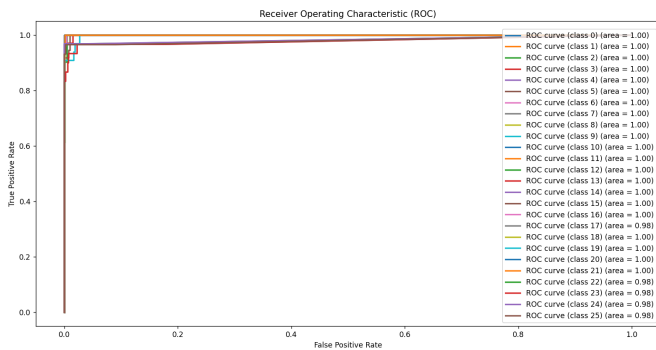


Fig. 8. ROC curve of SVM for Combined Dataset

Characteristic (ROC) curve for SVM, which provides the best outcome regarding its ability to distinguish different signs.

B. Real-time recognition

Fig. 9 shows a sample of real-time ASL signs with similar distances between landmarks but different orientations

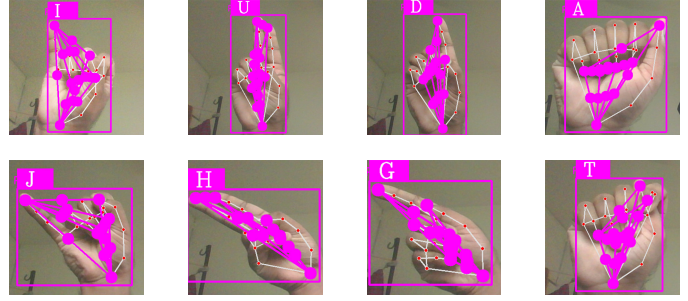


Fig. 9. Real-time sign recognition with different orientation

performed alongside their associated predicted letters in our system. The recognition system thus demonstrates the model's ability to comprehend precise hand gestures and shapes.

TABLE VI
TIME TAKEN FOR DIFFERENT MODELS

Algorithm	Training Time (Second)	Testing Time (Second)
KNN	0.03298	0.030983
SVM	1.5622	0.1644
RF	313.56	2.32
NB	0.007	0.002

Table VI shows the testing and training times for the different ML algorithms utilized in the study. Notably, NB is the fastest performer in training and testing, closely followed by KNN, demonstrating its efficiency in real-time sign prediction. While SVM requires significantly more time in training, it performs excellently in testing. RF, on the other hand, has a lengthier testing period due to its intrinsic complexity when compared to other algorithms. Our carefully prepared dataset, on the other hand, plays a critical role in enabling real-time sign prediction across all algorithms by reducing the system's processing load as well as its attributes show a close association with ASL alphabet signs. This is why, SVM along with other conventional algorithms performs with incredible accuracy and the ability to provide real-time predictions. Although SVM is not the fastest method, it is fast enough for real-time prediction, making it an excellent choice for practical applications for rapid and accurate detection.

V. CONCLUSION

ASL recognition, the process of interpreting American Sign Language gestures into corresponding alphabets, is facilitated through this study's vision-based approach for sign language recognition using hand landmarks and simplified Feature Extraction. Our dataset includes attributes linked to ASL alphabet signs, contributes to outstanding performance, with SVM emerging as the prevailing algorithm, achieving over 97% accuracy. Furthermore, investigating alternate approaches for distance measurement between landmarks, such as polar distance calculation, offers the possibility for significant advances in the recognition of sign languages. This study provides

promising rapid implications for ASL signers, allowing for enhanced communication and bridging the gap between ASL signers and those who are unfamiliar with sign language.

REFERENCES

- [1] "Deafness and hearing loss," Who.int. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss>. [Accessed: 28-Apr-2023].
- [2] D. A. Kent, "What is sign language?," University of Washington, 2013. [Online]. Available: <https://www.washington.edu/accesscomputing/what-sign-language>. [Accessed: 28-Apr-2023].
- [3] R. Rastgoo, K. Kiani, and S. Escalera, 'Sign Language Recognition: A Deep Survey', *Expert Systems with Applications*, vol. 164, p. 113794, 2021.
- [4] United Nations, "International Day of Sign Languages — United Nations," United Nations. [Online]. Available: <https://www.un.org/en/observances/sign-languages-day>. [Accessed: 28-Apr-2023].
- [5] M. S. Amin, S. T. H. Rizvi, and Md. M. Hossain, "A Comparative Review on Applications of Different Sensors for Sign Language Recognition," *Journal of Imaging*, vol. 8, no. 4, p. 98, Apr. 2022, doi: 10.3390/jimaging8040098.
- [6] N. Aloysius and M. Geetha, "Understanding vision-based continuous sign language recognition," *Multimed. Tools Appl.*, vol. 79, no. 31–32, pp. 22177–22209, 2020.
- [7] A. Halder and A. Tayade, 'Real-time vernacular sign language recognition using mediapipe and machine learning', *Journal homepage: www.ijrpr.com* ISSN, vol. 2582, p. 7421, 2021.
- [8] J. Bora, S. Dehingia, A. Boruah, A. A. Chetia, and D. Gogoi, 'Real-time Assamese Sign Language Recognition using MediaPipe and Deep Learning', *Procedia Computer Science*, vol. 218, pp. 1384–1393, 2023.
- [9] Q. Zhu, J. Li, F. Yuan, and Q. Gan, 'Continuous sign language recognition based on cross-resolution knowledge distillation', *ArXiv*, vol. abs/2303.06820, 2023.
- [10] "American sign language recognition and converter," *International Research Journal of Modernization in Engineering Technology and Science*, 2023.
- [11] A. Athania, K. Sanjay Gupta, and K. Khan, "Recognition of sign language in real time."
- [12] K. Bantupalli and Y. Xie, "American sign language recognition using deep learning and computer vision," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018, pp. 4896–4899.
- [13] R. Patil, V. Patil, A. Bahuguna, and G. Datkhile, "Indian Sign Language recognition using convolutional Neural Network," *ITM Web Conf.*, vol. 40, p. 03004, 2021.
- [14] M. J. Hossein and M. Sabbir Ejaz, "Recognition of Bengali sign language using novel deep convolutional neural network," in *2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 2020, pp. 1–5.
- [15] N. J. R. Muniraj, A. Poobalan, S. Pravin kumar, P. Sundar, and M. Surya, "Sign language recognition using artificial intelligence and machine learning," *Ijcert.org*. [Online]. Available: <https://ijcert.org/papers/IJCRT2205593.pdf>. [Accessed: 25-Mar-2023].
- [16] X. Jiang and W. Ahmad, "Hand gesture detection based real-time American sign language letters recognition using support vector machine," in *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCCom/CyberSciTech)*, 2019, pp. 380–385.