*Computers, Materials & Continua*
DOI: 10.32604/cmc.2022.025953
***Article***

Tech Science Press

# Intelligent Sign Language Recognition System for E-Learning Context

**Muhammad Jamil Hussain[1], Ahmad Shaoor[1], Suliman A. Alsuhibany[2], Yazeed Yasin Ghadi[3], Tamara al Shloul[4], Ahmad Jalal[1] and Jeongmin Park[5,*]**

[1]Department of Computer Science, Air University, Islamabad, 44000, Pakistan
[2]Department of Computer Science, College of Computer, Qassim University, Buraydah, 51452, Saudi Arabia
[3]Department of Computer Science and Software Engineering, Al Ain University, Al Ain, 15551, UAE
[4]Department of Humanities and Social Science, Al Ain University, Al Ain, 15551, UAE
[5]Department of Computer Engineering, Korea Polytechnic University, Siheung-si, 237, Gyeonggi-do, Korea
*Corresponding Author: Jeongmin Park. Email: jmpark@tukorea.ac.kr
Received: 09 December 2021; Accepted: 04 March 2022

**Abstract:** In this research work, an efficient sign language recognition tool for e-learning has been proposed with a new type of feature set based on angle and lines. This feature set has the ability to increase the overall performance of machine learning algorithms in an efficient way. The hand gesture recognition based on these features has been implemented for usage in real-time. The feature set used hand landmarks, which were generated using media-pipe (MediaPipe) and open computer vision (openCV) on each frame of the incoming video. The overall algorithm has been tested on two well-known ASL-alphabet (American Sign Language) and ISL-HS (Irish Sign Language) sign language datasets. Different machine learning classifiers including random forest, decision tree, and naïve Bayesian have been used to classify hand gestures using this unique feature set and their respective results have been compared. Since the random forest classifier performed better, it has been selected as the base classifier for the proposed system. It showed 96.7% accuracy with ISL-HS and 93.7% accuracy with ASL-alphabet dataset using the extracted features.

**Keywords:** Decision tree; feature extraction; hand gesture recognition; landmarks; machine learning; palm detection

## 1 Introduction

Human computer interaction (HCI) applications are now getting more and more attention from researchers around the world. This field deals with various interactions of humans with computers. Modern world applications require humans to interact with computers in many different ways to easily perform their everyday tasks [1,2]. Speech and vision signals are increasingly being used as inputs to many HCI speech recognition and computer vision-based applications. Human machine interaction (HMI) deals with the ways in which humans can interact with the machines. The machines are trained on speech data and therefore are usually interacted with using speech commands (e.g., Siri, Alexa).

The same logic applies to the systems trained on vision-based inputs (e.g., autonomous vehicles) [3]. In vision-based applications, the given scenes are understood intelligently by the machines, such as, in crowed anomaly systems [4] or human object interaction recognition systems [5]. Among vision-based inputs, there are many methods which humans use to interact with computers. These inputs are widely accepted and are most convenient when humans intend to give signals to machines. In these applications, human body shapes, facial expressions [6], hand gestures, and posture changes generate inputs for the systems to recognize and perform tasks [7] accordingly. This understanding or recognition of human gestures by machines is known as human gesture recognition (HGR) [8]. In noisy environments, such as, factory floors, the best way of interacting with machines is through hand gestures and vision-based systems. Also, there are some other environments, such as, hospitals and educational institutions where the speech inputs are inappropriate and can conflict with the regulations [9]. In these cases, the best and most convenient way to interact is through gestures [10]. Sign language through human hand gesture recognition is considered as one of the most effective ways in which human can interact with machines. This method is also useful for a community consisting of special individuals with speaking disabilities.

With these motivations, an intelligent sign language recognition tool has been developed to recognize hand gestures in real time [11]. The proposed system effectively extracts gestures from colored video frames. First, preprocessing is performed on every frame of the input video collected at the rate of 18 to 35 frames per second using webcam. Then, media-pipe hands pipelines detects palms and generates landmarks on hands. These landmarks are then used to extract the features which are discussed in detail in the following sections. The feature extraction process consists of two types of features based on angles and lines obtained from 21 landmarks. They are light weight in a sense that they do not require high computational power and ML classifiers are used to train the proposed model on these features, making it useful for real-time situations.

The rest of the paper is organized as follows: Section 2 describes the related research work and Section 3 presents the proposed methodology. Section 4 discusses the details of the used datasets and the performed experiments. The conclusion of the paper and the future work is discussed in Section 5.

## 2 Related Work

Hand gesture recognition is not a new idea in the field of HCI. There are two types of gestures: static and dynamic gestures [12]. Gestures that remain the same for a given time frame are called static gestures while dynamic gestures keep changing in a given time frame, such as, waving. Some common approaches for hand gesture recognition include the use of electromyography (EMG) [13], cameras [14], and wearable gloves [15]. Asif et al. [16] used a convolutional neural network (CNN) to detect and recognize hand gestures. After 80–100 epochs, the system begins to understand the gestures from the superficial electromyography (sEMG) data recorded from 18 subjects. In [17], a signal processing and sliding window approach was used to recognize hand gestures through electromyography (EMG) and artificial neural network (ANN) with an accuracy of 90.7%. In [18], a real-time hand gesture recognition model was introduced that collected data from a Myo armband worn on the subject's forearm. The model is based on k-nearest neighbor (KNN) and dynamic time warping algorithms. The model performed with a high accuracy of 86% with 5 classes of gestures. In [19], a unique approach was presented to minimize the unnecessary or redundant information of EMG and also to increase the performance of real-time recognition of hand gestures through principal component analysis (PCA) and generalized regression neural network (GRNN) with 95.1% accuracy. The recognition of hand gestures through ultrasonic sensors and smart phones has attracted the attention of many researchers

during the past few years [20]. An online machine learning solution, known as "wisture", can recognize hand gestures on smartphones. It has been trained using a recurrent neural network (RNN) with an accuracy of up to 93% with three hand gestures [21]. Panella et al. [22] discussed the problems in recognizing the hand gestures through hand segmentation in devices like smart phones with less resources. They introduced a new and efficient ML algorithm, which is capable of recognizing the hand gestures through Hu image moments [23].

Extracting features from hand is quite a difficult task. Numerous researchers proposed different methods and techniques for this purpose. Oprisescu et al. [24] suggested that the gestures input should be taken from depth and time-of-flight (Tof) cameras. The gestures are recognized using the decision tree classifier and an accuracy of 93.3% was achieved on 9 different gestures. Yun et al. [25] proposed a multi-feature fusion and template matching technique for classification of hand gestures. Ahmed et al. [26] used dynamic time wrapping to recognize 24 hand gestures of Indian sign language with an accuracy score of 90%. Pansare et al. [27] proposed an alphabet American sign language recognizer (A-ASLR) based on real-time vision and the ASL alphabets dataset which obtained an accuracy of 88.26%. Ansar et al. [28] used a point-based full hand-based feature extraction method. Gray wolf optimizer was used to optimize features and genetic algorithm was used to classify hand gestures. Various landmark extraction techniques are being developed these days. Shin et al. [29] extracted features using 21 hand landmarks. These features were angle and distance based through landmark extracted through MediaPipe (media-pipe). A combination of support vector machine (SVM) and light gradient boosting machine (GBM) was used to classify the hand gestures. Costa et al. [30] also used media-pipe for landmark extraction but they obtained bounding boxes around the hands by selecting the top and bottom landmarks. For classification, they used SVM and achieved 90% accuracy. However, the methods which are distance and area based become problematic when tested in real-world situations using 2D cameras. There is a high chance that there would be very diverse feature values for distance and area-based methods for a single gesture.

## 3 Method and Materials

This section explains the proposed methodology in detail. Apart from the two datasets, a camera has also been used to record hand gestures and the obtained videos have been converted into image frames. The frames have been passed through a pre-processing phase using open computer vision (openCV) library which reads the video frame by frame. These frames have been sent to media-pipe which locates landmarks in each frame. The feature extraction module has used these landmarks to extract features. Then the extracted feature set has been passed to train classifiers. An overview of the system is shown in Fig. 1.

### *3.1 Preprocessing*

The OpenCV library [31] has been used to read the dataset image frames. The ASL-alphabet (American Sign Language) and ISL-HS (Irish Sign Language) datasets has been selected for this work. A publicly available pipeline known as MediaPipe [32] has been used for generation of landmarks on hands. This pipeline uses two models: a palm detection model that works on the whole frames and returns the region of interest (ROI). The ROI contains the cropped hand in the returned frame from detection model. It becomes input for landmark model (also known as joint locator model) which works on this ROI and generates 3D landmarks on hand via regression. The MediaPipe models are trained on 30000 colored images. This pipeline returns the values of pixels on which landmarks are located by models, i.e., x and y coordinates on image as shown in Fig. 2.

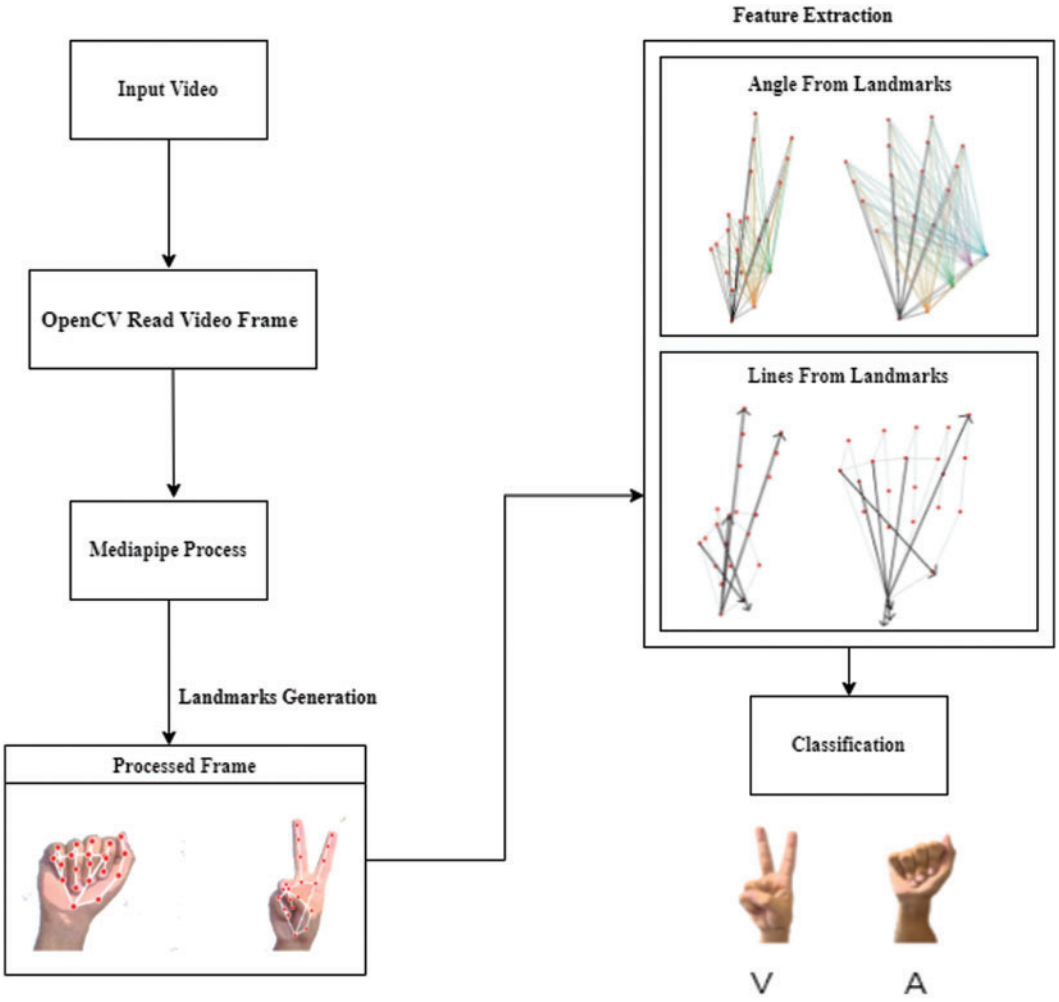



**Figure 1:** A general overview of the proposed system design

Since21 landmarks are generated on each hand, each landmark could be accessed via a specific id from 0 to 20. The MediaPipe assigns a token or id on each landmark as shown in Fig. 3. These landmarks can help extract different features including the distance and angle between any two landmarks.

### *3.2 Features Extraction*

The angles and lines are the feature extracted in feature extraction module.

#### *3.2.1 Angle Features*

For extracting the angle features, the slopes between each pair of landmarks are calculated using Eq. (1).

$$S_{i,j} = \frac{y_j - y_i}{x_j - x_i} \tag{1}$$

where $(x_i, y_i)$, $(x_j, y_j)$ is a pair of landmarks and $S_{i,j}$ is the slope between them. These slopes are used to calculate the angles of landmarks using Eq. (2).

$$\theta_{i,j} = tan^{-1}\left(S_{i,j}\right) \tag{2}$$

where $\theta_{i,j}$ is the angle between the slope and the x-axis. For every slope, the angle is calculated using the above equation in the proposed method. Fig. 4 shows the calculation of 3 different angles.

(a) (b) (c)

**Figure 2:** Media-pipe results on (a) Closed fist as "A" gesture, (b) **"**V" gesture and (c) "P" gesture

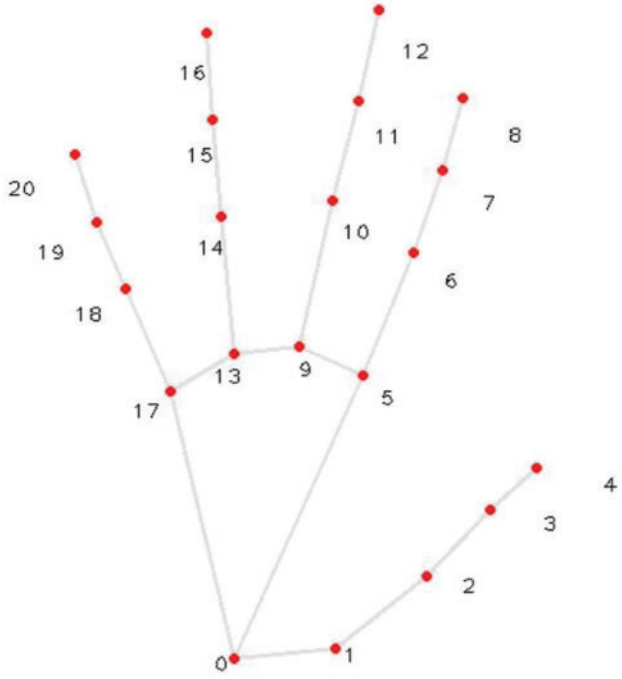



**Figure 3:** Landmark generation on hand and finger from 0 to 20

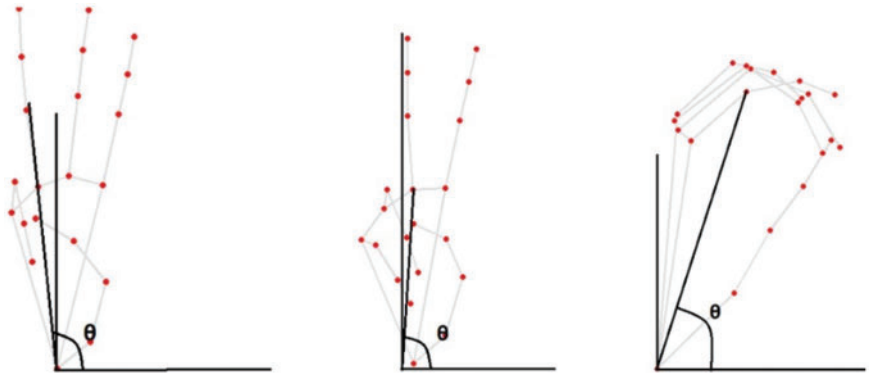



**Figure 4:** Angle feature representations including (a) Angles between landmarks 0 and 5, (b) Angles between landmarks 0 and 9 and (c) Angles between landmarks 0 and 6

### *3.2.2 Lines Features*

The fingers are considered as lines in this type of feature extraction. The fingers are labeled from 0 to 4 as shown in Fig. 5. The slopes of fingers are calculated using bottom and top landmarks for each finger using Eq. (1).

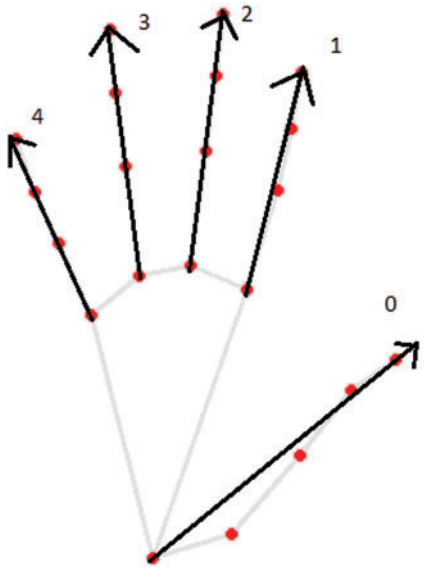



**Figure 5:** Line representation of 5 fingers

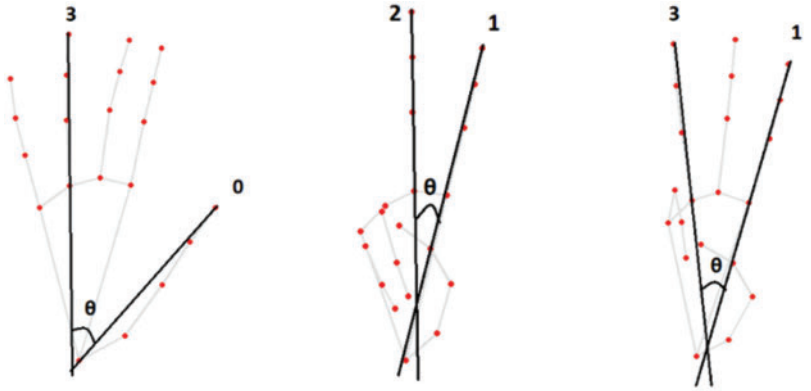
Angles between fingers are calculated based on slopes of fingers using Eq. (3). These angles between fingers are used as new features. Fig. 6 shows the fingers as lines. Eq. (3) is used for calculation of angle between lines $i$ and $j$.

$$\theta_{i,j} = \left| \frac{S_i - S_j}{1 + S_i S_j} \right| \tag{3}$$

where $\theta_{i,j}$ is the angle between the slopes $S_i$ and $S_j$ (slopes of finger $i$ and $j$). The value of $i$ and $j$ is 0, 1, 2, 3, 4 representing fingers.

**Algorithm 1:** Extraction of angle and line features between every pair of landmarks

**Input:** dataset ASL-alphabet/ISL-HS
**Output:** Extracted feature from hand pose
**for** n = 0 : m ( for every frame)
**1** Take each Frame and generate landmarks via media pipe.
**for** i = 0: p (for every possible pair)
**2** Compute slopes between a pair$_i$ of land mark. Using Eq. (1)
**3** Take slopes and compute angle via Eq. (2) and store it.
**4** Take slopes and calculate lines via Eq. (3) and store it.
**end**
**end**
**return( )** Extracted Features of angles and lines




(a) (b) (c)

**Figure 6:** Angles between different lines, (a) Angle between lines 0 and 3 (b) Angle between lines 1 and 2 and (c) Angle between lines 1 and 3

After extracting the angle and line features, a dimensionality reduction technique is applied [33]. More than half of the dimensionality of the feature vector is reduced in this step. Features were highly correlated (i.e., angle from landmark 0 to landmark 1 and angle from landmark 1 to landmark 0).

### *3.3 Model Training*

This feature extraction module is applied to ASL-alphabet and ISL-HS datasets for feature extraction. The generated features are loaded for model training. These ML classifiers are selected over deep learning methods because they are more interpretable. Interpretability is the important factor in this research work since we want to investigate the performance state-of-art ML classifiers with the proposed feature set.

### *3.4 Classifiers*

The methodology is evaluated using different machine learning classifiers such as naïve Bayes, decision trees, and random forest.

### 3.4.1 Naïve Bayes

The naïve Bayes is the first classifier used to train the model. It works best on independent features. For example, if feature B has a certain value, then it determines the probability of class 'A' using Eq. (4).

$$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right) P(A)}{P(B)} \tag{4}$$

In our case, B is the feature set (angle and line values) and A is the gesture. Since B is the features set ( $f_1$, $f_2, f_3, f_4, f_5 \ldots f_{440}$ ) according to the proposed scenario, Eq. (4) can be rewritten for class A as Eq. (5).

$$P\left(\frac{A}{f_1, f_2, f_3, f_4, f_5 \ldots f_{440}}\right) = \frac{P\left(\frac{f_1}{A}\right).P\left(\frac{f_2}{A}\right).P\left(\frac{f_3}{A}\right).P\left(\frac{f_4}{A}\right)\ldots P\left(\frac{f_{440}}{A}\right).P(A)}{P(f_1).P(f_2).P(f_3)\ldots P(f_{440})} \tag{5}$$

### 3.4.2 Decision Tree

The decision tree takes features as nodes of a tree called decision nodes. The leaf nodes of a decision tree are the classes. The root node is the feature with the highest information gain to reduce the interclass similarities. For every class, there is a different path followed on the tree to give accurate output. The decision at each node is dependent upon the value of the nodes or features traversed (e.g., the value of F3 is less than equal to 1.2 and the next decision node is F2) for a particular gesture as shown in Fig. 7.

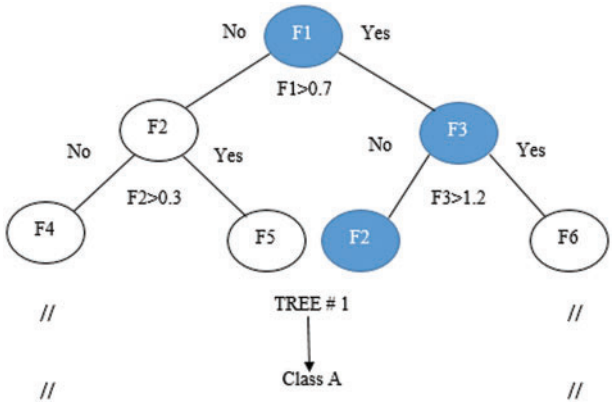



**Figure 7:** Decision tree with feature set <F1, F2, F3, F4, and F6>, F1 is the root node with value 0.7

Information gain for root node can be calculated using Eq. (6).

$$Gain = Entropy(root) - \sum_{i \in childs(root)} \frac{|i|}{|root|} Entropy(i) \tag{6}$$

where gain is the information gain for node or split and $i$ is the child of the root node. It is calculated by subtracting entropy of all child nodes from the entropy of the node for which the information is being calculated (root node in above Eq. (6)). Entropy measures the purity of split. The higher the entropy, the harder it is to draw any conclusion from the available information. The entropy of a node

can be calculated using Eq. (7)

$$Entropy\,(node) = -\sum_{i=1}^{C} p_i \log_2 p_i \tag{7}$$

where c represents a class and $p_i$ represents the probability of that class with respect to the node on which the entropy is being calculated.

#### *3.4.3 Random Forest*

Random forest classifier is selected as the base classifier for classification of hand gestures in the proposed method. The random forest consists of multiple decision trees and in this approach, the tree count is 100. It is an ensemble-based classifier which does majority voting after collecting the recognized labels from each decision tree against a given sample.

## 4 Experimental Setting and Results

This section gives a brief description of the performance of the proposed system using five different classifiers on the two datasets. All these experiments were performed using pycharm integrated development environment (ide) for python. The material provided in this section of the paper will validate the system. First, it provides the description of the datasets and then gives the details of the various experiments conducted to establish the high performance of the proposed system on the basis of accuracy, precision, recall score, F1 score, learning curves, and heat maps.

### *4.1 Datasets Description*

The two publically available well known datasets for sign language are selected. These include ASL alphabet dataset [34] and ISL-HS dataset [35].

#### *4.1.1 ASL Alphabet Dataset*

This dataset [17] contains 28 gestures: the alphabets of the English language (A, B ... Y, Z) and the two signs of space and delete. The gestures are single handed (from right hand) performed under similar lighting conditions and backgrounds. Performing real-time hand gesture recognition on this dataset is challenging because there is high interclass similarity between different classes. The dataset contains 3000 colored images per gesture (a total of 87000 images for all 28 gestures). Few samples of this dataset as shown in Fig. 8.

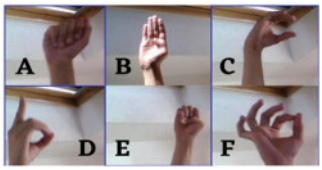



**Figure 8:** Samples of ASL alphabet dataset [34] includes gestures of A, B, C, D, E and F

#### *4.1.2 ISL Dataset*

This dataset [18] contains hand gestures for 26 alphabets of the English language. Every sign has 18 videos, so there are a total of 468 colored videos for all gestures and each of them is roughly 3 s long. We have taken only the first 60 frames from every video to generate features. The rationale behind this was

that the orientation of landmarks will change with time between the samples of dynamic datasets and produce diverse values for a single gesture or class. This effects the overall performance of algorithm. Therefore, only the first 60 frames are taken so that orientation remains almost the same for every sample. Few samples of this dataset are available as shown in Fig. 9.

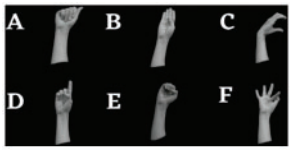



**Figure 9:** Samples of ISL-HS dataset [35] includes gestures of A, B, C, D, E and F

### *4.2 Results*

#### *4.2.1 Learning Curves*

With 10-fold cross validations, the learning curves of different classifiers are plotted. Fig. 10 represents the learning curves for three different classifiers on ISL-HS of random forest; namely, random forest, decision tree and naïve Bayes.

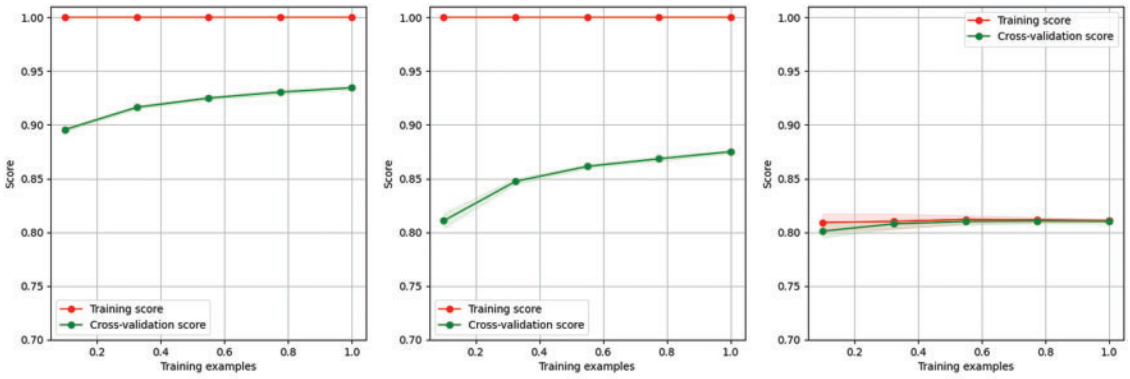



**Figure 10:** Accuracy score on y-axis and training examples on x-axis for (a) Random forest curve, (b) Decision tree curve and (c) Naive bayes curve

The cross-validation score in each of the curves is low and increases with time. For random forest, the curve shows very high performance as compared to the decision tree and the naïve Bayes classifier. The training score is nearly maximum in random forest and decision tree but it decreases over time in case of naïve Bayes. The learning curves for ASL-alphabet dataset as shown in Fig. 11.

The cross-validation score in each of the curves is low and increases with time. For random forest, the curve shows very high performance as compared to the decision tree and the naïve Bayes classifier. The training and cross-validation scores of the models are low on the ASL dataset as compared to the ISL dataset. This is because there is high interclass similarity in the ASL dataset as compared to the ISL dataset.

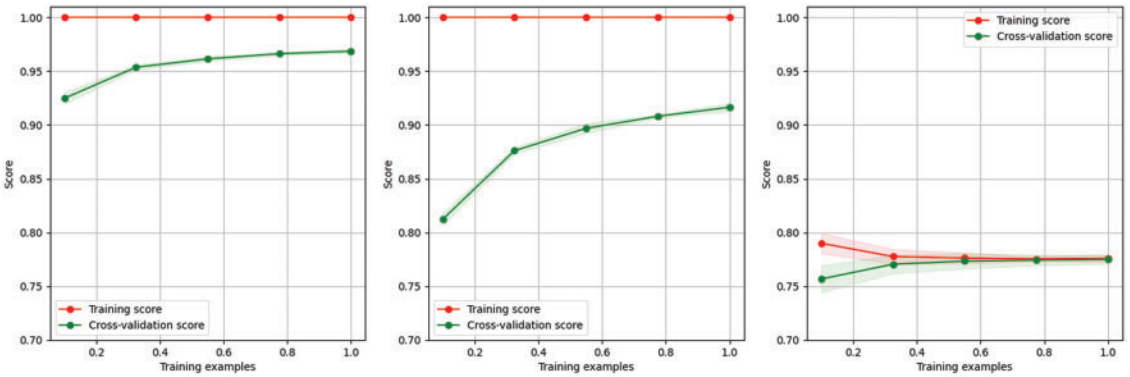



**Figure 11:** Accuracy score on y-axis and training examples on x-axis for (a) Random forest curve, (b) Decision tree curve and (c) Naive bayes curve

### *4.2.2 Recognition Accuracy, Precision, Recall, F1*

Tabs. 1 and 2 show the accuracy, precision, recall, and F1 scores obtained using five different classifiers; namely, Naïve Bayes, decision tree, random forest, k-nearest neighbors (KNN) and support vector machine (SVM) over the ISL-HS and ASL alphabet datasets respectively.

**Table 1:** Precision, accuracy, recall and F1 score of different classifiers over the ISL dataset

| | Naïve bayes | Decision tree | Random forest | KNN | SVM |
|---|---|---|---|---|---|
| **Precession** | 0.793 | 0.915 | 0.968 | 0.948 | 0.957 |
| **Accuracy** | 0.780 | 0.916 | 0.968 | 0.949 | 0.957 |
| **Recall** | 0.783 | 0.916 | 0.968 | 0.949 | 0.957 |
| **F1 score** | 0.782 | 0.915 | 0.968 | 0.948 | 0.957 |

**Table 2:** Precision, accuracy, recall and F1 score of different classifiers over the ASL dataset

| | Naïve bayes | Decision tree | Random forest | KNN | SVM |
|---|---|---|---|---|---|
| **Precession** | 0.84 | 0.872 | 0.934 | 0.897 | 0.924 |
| **Accuracy** | 0.81 | 0.878 | 0.937 | 0.901 | 0.925 |
| **Recall** | 0.80 | 0.872 | 0.932 | 0.896 | 0.921 |
| **F1 score** | 0.81 | 0.872 | 0.932 | 0.896 | 0.922 |

From Tabs. 1 and 2, it is conclusive that the random forest classifier works better with the proposed algorithm as compared to the other four classifiers. Also, it can be seen that Naïve Bayes achieved the lowest accuracies because it works best with independent features. In the proposed method, however, the features are interdependent. For example, when the angle between landmark 0 and landmark 12 (12 is on the tip of the middle finger) is changed, then its effect is also reflected over the other features in a way that the angle between landmark 12 and landmark 4 is also changed. Tabs. 3 and 4 show the confusion matrices depicting the accuracies of the selected ASL alphabet and ISL-HS datasets respectively.

**Table 3:** Confusion matrix of random forest classifier on ASL dataset

| | A | B | C | D | de | E | F | G | H | I | J | K | L | M | N | nt | O | P | Q | R | S | sp | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | **0.94** | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B | 0 | **0.97** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| C | 0 | 0 | **0.97** | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D | 0 | 0 | 0 | **0.97** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| de | 0 | 0 | 0.01 | 0.01 | **0.87** | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0.02 | 0.01 | 0 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 |
| E | 0 | 0 | 0 | 0 | 0 | **0.94** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F | 0 | 0 | 0 | 0.01 | 0 | 0 | **0.94** | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| G | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.96** | 0.01 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| H | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | **0.97** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | **0.93** | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| J | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | **0.96** | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 |
| K | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.96** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0 | 0.01 | 0.01 | 0 | 0.01 | 0 | 0 |
| L | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.96** | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M | 0.01 | 0.01 | 0 | 0 | 0.01 | 0.01 | 0.01 | 0 | 0 | 0.01 | 0 | 0 | 0 | **0.87** | 0.04 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.14 | **0.8** | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| nt | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| O | 0 | 0 | 0.01 | 0.02 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | **0.93** | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| P | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.94** | 0.02 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Q | 0 | 0 | 0 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | **0.93** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.93** | 0 | 0.01 | 0 | 0.02 | 0 | 0 | 0.01 | 0 | 0 |
| S | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | **0.94** | 0.01 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 |
| sp | 0 | 0 | 0 | 0.01 | 0.01 | 0.01 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.03 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | 0 | **0.85** | 0 | 0 | 0 | 0 | 0.01 | 0.01 | 0 |
| T | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.96** | 0 | 0 | 0 | 0 | 0 | 0 |
| U | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0.01 | 0 | **0.92** | 0.01 | 0 | 0.01 | 0 | 0 |
| V | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.02 | 0 | 0.01 | **0.91** | 0 | 0.01 | 0.01 | 0 |
| W | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | **0.94** | 0 | 0 | 0 |
| X | 0 | 0 | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.02 | 0 | 0.01 | 0 | 0 | **0.91** | 0 | 0 |
| Y | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | **0.95** | 0 |
| Z | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.98** |

Note: De = del; nt = nothing; sp = space;

**Table 4:** Confusion matrix of random forest classifier on ISL dataset

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B | 0 | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| C | 0 | 0 | **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D | 0 | 0 | 0 | **0.98** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| E | 0 | 0 | 0 | 0 | **0.98** | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F | 0 | 0 | 0 | 0 | 0 | **0.98** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| G | 0 | 0 | 0 | 0 | 0 | 0.01 | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.98** | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.99** | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| J | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0.01 | 0.01 | **0.95** | 0 | 0 | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 |
| K | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| L | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | **0.94** | 0.05 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.12 | **0.78** | 0.08 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 |
| O | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.14 | **0.85** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| P | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

(Continued)

**Table 4:** Continued

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Q | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.98** | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 |
| S | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.99** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| T | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.98** | 0 | 0 | 0 | 0 | 0 | 0 |
| U | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | **0.97** | 0 | 0 | 0 | 0 | 0 |
| V | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | **0.98** | 0 | 0 | 0 | 0 |
| W | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.98** | 0 | 0 | 0 |
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **1** | 0 | 0 |
| Y | 0.01 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0.96** | 0 |
| Z | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | **0.96** |

In Figs. 12 and 13, the results are validated through heat maps for ASL-alphabet and ISL-HS datasets respectively.

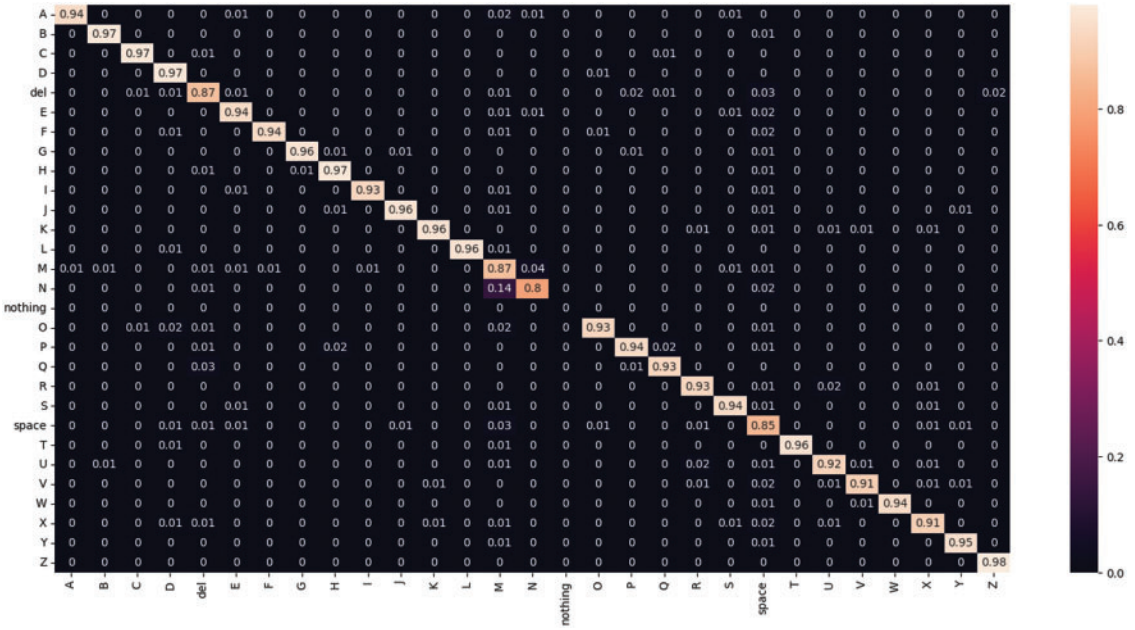



**Figure 12:** Heat map for ASL-alphabet dataset

While in Tab. 5, the proposed system has been compared with other state-of-the-art methods and has shown the most promising results.

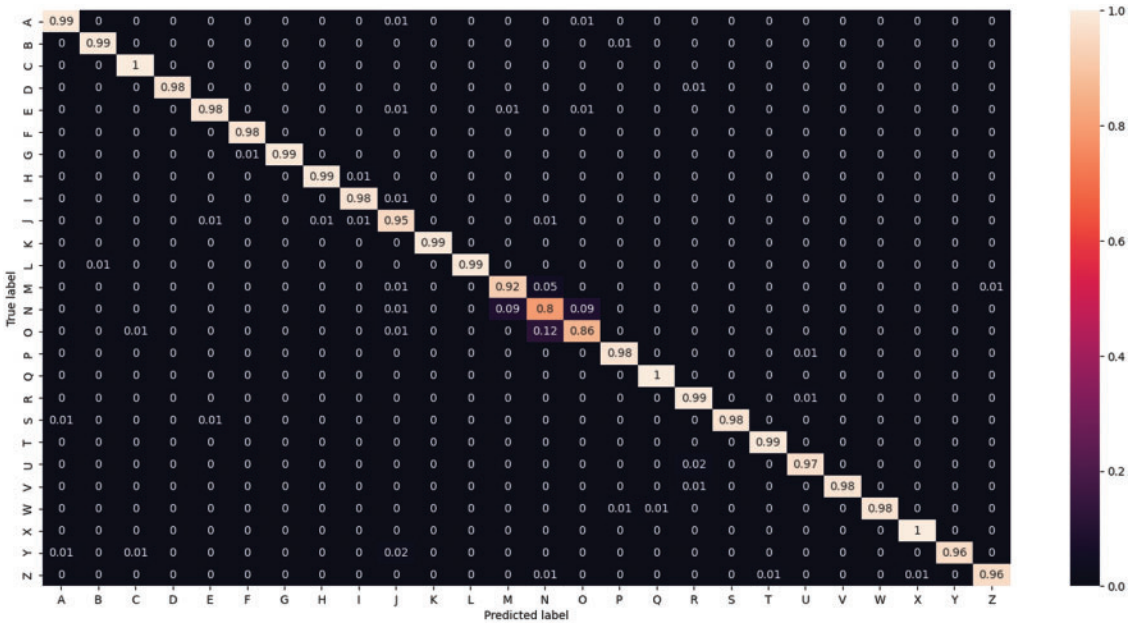



**Figure 13:** Heat map for ISL-HS dataset

**Table 5:** Comparison of the proposed method with other state-of-the-art methods on datasets

| Datasets | Methods | Recognition accuracy (%) |
|---|---|---|
| ASL | Light GBM [29] | 86.12 |
| | SVM [29] | 87.6 |
| | Bonding box around hands [30] | 90 |
| | Pruned DCNN [36] | 91 |
| | SqueezeCapsNet [37] | 91.6 |
| | **Proposed methods** | **93.7** |
| ISL-HS | Phase 2 VGGNet16 [38] | 71.4 |
| | Non-blur (PCA + k-NN) HORF [39] | 86.85 |
| | Non-blur (PCA + k-NN) IRF [39] | 94.6 |
| | Non-blur PCA [40] | 95 |
| | **Proposed method** | **96.8** |

## 5 Conclusion

In this research work, an efficient system has been proposed which performed well on ASL alphabet and ISL-HS datasets. Among the five chosen classifiers, andom forest has shown the highest accuracy on the selected datasets. It achieved an accuracy of 93% on ASL alphabet dataset and 96.7% on ISL dataset. The proposed system is light weight and will perform well in variable environments. However, it shows limitations when it comes to dynamic datasets since the orientations of hand

landmarks with each other in multiple frames of a dynamic class sample generate very different values for a single hand gesture label.

The authors are planning to add more features in addition to angles and lines to increase the system's performance for dynamic and versatile environments. Moreover, they are also trying to devise a strategy to handle dynamic datasets.

**Funding Statement:** This research was supported by a Grant (2021R1F1A1063634) of the Basic Science Research Program through the National Research Foundation (NRF) funded by the Ministry of Education, Republic of Korea.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] S. Diederich, A. B. Brendel, S. Morana and L. Kolbe. "On the design of and interaction with conversational agents: An organizing and assessing review of human-computer interaction research," *Journal of Association for Information Systems*, vol. 23, no. 1, pp. 96–138, 2022.

[2] J. Zhou. "Deep learning-driven distributed communication systems for cluster online educational platform considering human–computer interaction," *International Journal of Communication Systems*, vol. 35, no. 1, pp. 1–15, 2022.

[3] I. Ahmad and K. Pothuganti, "Design & implementation of real time autonomous car by using image processing & IoT," in *Proc. Int. Conf. on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, pp. 107–113, 2020.

[4] A. Faisal, J. Madiha and A. Jalal, "Crowd anomaly detection in public surveillance via spatio-temporal descriptors and zero-shot classifier," in *Proc. IEEE Int. Conf. on Innovative Computing*, Islamabad, Pakistan, pp. 1–6, 2021.

[5] N. Khalid, Y. Y. Ghadi, M. Gochoo, A. Jalal and K. Kim, "Semantic recognition of human-object interactions via gaussian-based elliptical modeling and pixel-level labeling," *IEEE Access*, vol. 9, pp. 111249–111266, 2021.

[6] H. Ge, Z. Zhu, Y. Dai, B. Wang and X. Wu. "Facial expression recognition based on deep learning," *Journal of the Computer Methods and Programs in Biomedicine*, vol. 4, pp. 106621, 2022.

[7] O. Patsadu, C. Nukoolkit and B. Watanapa, "Human gesture recognition using kinect camera," in *Proc. Joint Conf. on Computer Science and Software Engineering (JCSSE)*, Bangkok, Thailand, pp. 28–32, 2012.

[8] S. Ahlawat, V. Batra, S. Banerjee, J. Saha and A. K. Garg, "Hand gesture recognition using convolutional neural network," in *Proc. Int. Conf. on Innovative Computing and Communication (ICICC)*, Singapore, pp. 179–186, 2019.

[9] B. Hariharan, S. Padmini and U. Gopalakrishnan, "Gesture recognition using kinect in a virtual classroom environment," in *Proc. Digital Information and Communication Technology and its Applications (DICTAP)*, Bangkok, Thailand, pp. 118–124, 2014.

[10] W. Li, C. Hsieh, L. Lin and W. Chu, "Hand gesture recognition for post-stroke rehabilitation using leap motion," in *Proc. Industry Consortium for Advancement of Security on the Internet (ICASI)*, Sapporo, Japan, pp. 386–388, 2017.

[11] M. J. Hussain, "Video for real time hand gesture recognition," [Online]. Available: https://www.youtube.com/watch?v=xMXxg_eudvg. 2022.

[12] S. Joudaki and A. Rehman, "Dynamic hand gesture recognition of sign language using geometric features learning," *International Journal of Computational Vision and Robotics*, vol. 12, pp. 1–16, 2022.

[13] Z. Zhang, K. Yang, J. Qian, and L. Zhang, "Real-time surface EMG pattern recognition for hand gestures based on an artificial neural network," *Sensors*, vol. 19, no. 14, pp. 3170, 2019.

[14] M. Oudah, A. Al-Naji and J. Chahl, "Hand gestures for elderly care using a microsoft kinect," *Nano Biomedicine and Engineering*, vol. 12, no. 3, pp. 197–204, 2020.
[15] F. Pezzuoli, D. Corona and M. L. Corradini, "Recognition and classification of dynamic hand gestures by a wearable data-glove," *SN Computer Science*, vol. 2, no. 1, pp. 1–9, 2021.
[16] A. R. Asif, A. Waris, S. O. Gilani, M. Jamil, H. Ashraf *et al.,* "Performance evaluation of convolutional neural network for hand gesture recognition using EMG," *Sensors*, vol. 20, pp. 1642, 2020.
[17] C. Motoche and M. Benalcázar, "Real-time hand gesture recognition based on electromyographic signals and artificial neural networks," in *Proc. Int. Conf. on Artificial Neural Networks*, Rome, Italy, pp. 352–361, 2018.
[18] M. Benalcázar, A. Jaramillo, A. Jonathan, A. Paez and V. Andaluz, "Hand gesture recognition using machine learning and the Myo armband," in *Proc. 25th European Signal Processing Conf. (EUSIPCO)*, Kos, Greece, pp. 1040–1044, 2017.
[19] J. Qi, G. Jiang, G. Li, Y. Sun and B. Tao, "Surface EMG hand gesture recognition system based on PCA and GRNN," *Neural Computing and Applications*, vol. 32, pp. 6343–6351, 2020.
[20] Z. Wang, Y. Hou, K. Jiang, W. Dou, C. Zhang *et al.,* "Hand gesture recognition based on active ultrasonic sensing of smartphone: A survey," *IEEE Access*, vol. 7, pp. 111897–111922, 2019.
[21] M. A. A. Haseeb and R. Parasuraman, "Wisture: Touch-less hand gesture classification in unmodified smartphones using Wi-Fi signals," *IEEE Sensors Journal*, vol. 19, no. 1, pp. 257–267, 2019.
[22] M. Panella and R. Altilio, "A Smartphone-based application using machine learning for gesture recognition: Using feature extraction and template matching via Hu image moments to recognize gestures," *IEEE Consumer Electronics Magazine*, vol. 8, no. 1, pp. 25–29, 2019.
[23] M. -K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.
[24] S. Oprisescu, C. Rasche and B. Su, "Automatic static hand gesture recognition using ToF cameras," in *Proc. 20th European Signal Processing Conf. (EUSIPCO)*, Bucharest, Romania, pp. 2748–2751, 2012.
[25] L. Yun, Z. Lifeng and Z. Shujun, "A hand gesture recognition method based on multi-feature fusion and template matching," in *Proc. Eng. 29*, Harbin, China, pp. 1678–1684, 2012.
[26] W. Ahmed, K. Chanda and S. Mitra, "Vision based hand gesture recognition using dynamic time warping for Indian sign language," in *Proc. Int. Conf. on Information Science (ICIS)*, Dublin, Ireland, pp. 120–125, 2016.
[27] J. R. Pansare and M. Ingle, "Vision-based approach for American sign language recognition using edge orientation histogram," in *Proc. Int. Conf. on Image, Vision and Computing (ICIVC)*, Portsmouth, UK, pp. 86–90, 2016.
[28] H. Ansar, A. Jalal, M. Gochoo and K. Kim, "Hand gesture recognition based on auto-landmark localization and reweighted genetic algorithm for healthcare muscle activities," *Sustainability*, vol. 13, no. 5, pp. 2961, 2021.
[29] J. Shin, A. Matsuoka, M. A. M. Hasan and A. Y. Srizon, "American sign language alphabet recognition by extracting feature from hand pose estimation," *Sensors*, vol. 21, no. 17, pp. 5856, 2021.
[30] A. Costa and U. Edu, "ASLScribe: Real-time American sign language alphabet image classification using mediapipe hands and artificial neural networks," final project, University of Georgia, 2019.
[31] "OpenCV," [Online]. Available: https://opencv.org/. 2021.
[32] "Mediapipe," [Online]. Available: https://google.github.io/mediapipe/solutions/hands. 2020.
[33] M. Javeed, A. Jalal and K. Kim, "Wearable sensors based exertion recognition using statistical features and random forest for physical healthcare monitoring," in *Proc. Int. Bhurban Conf. on Applied Sciences and Technologies (IBCAST)*, Islamabad, Pakistan, pp. 512–517, 2021.
[34] Akash, "ASL alphabet," [Online]. Available: https://www.kaggle.com/grassknoted/asl-alphabet. 2018.
[35] M. Oliveira, H. Chatbri, Y. Ferstl, M. H. Farouk, S. Little *et al.,* "A dataset for Irish sign language recognition," in *Proc. Irish Machine Vision and Image Processing (IMVIP)*, Maynooth, Ireland, 2017.

[36] A. Ashiquzzaman, H. Lee, K. Kim, H. -Y. Kim, J. Park *et al.,* "Compact spatial pyramid pooling deep convolutional neural network based hand gestures decoder," *Applied Sciences*, vol. 10, no. 21, pp. 7898, 2020.

[37] S. Sridhar and S. Sanagavarapu, "Squeezecapsnet–transfer learning-based ASL interpretation using squeezeNet with multi-lane capsules," in *Proc. IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conf. (UEMCON)*, New York, NY, USA, pp. 1–7, 2021.

[38] M. Oliveira, H. Chatbri, N. Yarlapati, N. E. Connor and A. Sutherland, "Hand orientation redundancy filter applied to hand-shapes dataset," in *Proc. 2nd Int. Conf. on Applications of Intelligent Systems*, Canaria Spain, pp. 1–5, 2019.

[39] F. Fowley and A. Ventresque, "Sign language fingerspelling recognition using synthetic data," in *Proc. 29th Irish Conferenfe on Artificial Intelligence and Cognitive Science*, Dublin, Ireland, pp. 1–6, 2021.

[40] M. Oliveira, H. Chatbri, S. Little, Y. Ferstl and N. E. O'Connor *et al.,* "Irish sign language recognition using principal component analysis and convolutional neural networks," in *Proc. Digital Image Computing: Techniques and Applications (DICTA)*, Sydney, Australia, pp. 1–8, 2017.