Hasta mudra abhijñānam

Indian sign language recognition

[1]Shushma Khanvilkar, [2]Neha Kesarkar, [3]Oswyn Lewis, [4]Sagar Mehta

[1]Project guide, [2]Student, [3]Student, [4]Student,
[1]Computer Engineering,
[1]Xavier Institute of Engineering, Mumbai, India.

Abstract: This paper introduces a novel system to recognize Indian Sign Language (ISL) in Real-Time through the mobile application. This system will bridge the communication gap between the hearing, speech impaired and the rest of the society. The existing systems provide low accuracy and may not work in real-time. Our system provides accurate results and works in real-time. ISL signs and gestures will be captured through a phone camera and these gestures/signs will be decrypted into English text. The use of external hardware like sensors, gloves, etc. is not required. In India, there are 15 million people who cannot hear or speak. Also, to avoid the expense required for sensor-based systems, our application will be helpful for impaired people to communicate with society. Since this system is application-based, it can be used in all sectors of society.

Index Terms* - *Real time, Android application, Indian sign language, Computer vision, Machine learning, K-Nearest Neighbor, Gesture recognition, Grid-based feature extraction

I. INTRODUCTION

Indian Sign Language (ISL) is a commonly used language among deaf and dumb people in India. ISL consists of different signs (mudras) and a combination of signs is called a gesture. Most of the society cannot understand Indian sign language and hence they need interpreters while communicating to impaired people. However, one cannot find interpreters to translate these signs or gestures. This research presented in the paper suggests a system that will help comprehend Indian Sign Language so that the need for an interpreter in daily conversations will not be required & the impaired people will be able to easily communicate with society.

In today's world, smartphones are easily affordable. Our system being application-based can be used by everyone. This application can be used in the education sector so that the need for special schools or colleges will be eradicated. Also, the application can help the impaired people find employment in offices where they can communicate with other employees by using the ISL application.

Our system comprises an android smartphone camera to capture hand poses and gestures and our server processes frames that are received by the smartphone camera. The system can identify all 26 hand poses in ISL, which include all alphabets. The research presented pertains to Indian sign language defined in Talking Hands website [1]. The hand poses in ISL, as shown in Fig. 1, are used to train the machine.

In the next section of the paper, the present systems pertaining to sign language recognition are discussed. Section III explains the flow and implementation of the system. Section IV describes the application developed for the system to recognize ISL in Real-time. In section V, we will discuss the future scope of the system. In the last section i.e., section VI the success of the implementation is discussed.

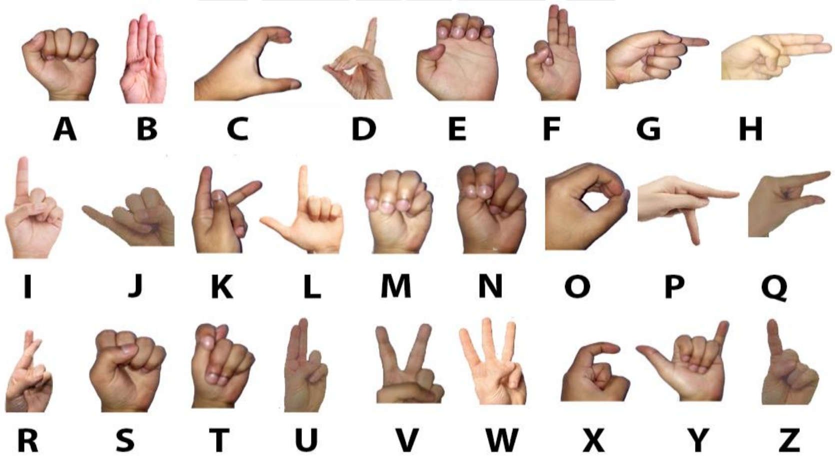

Fig (1): alphabets in Indian Sign Language.

II. Existing System

Sign language (SL) is an important field of research in the field of computer vision. SL is a physical action by using hands and eyes with which we can communicate with dumb and deaf people. The impaired people can express their feelings with various hand moments and poses. Each country has its own Sign language. Considerable work has been done in the field of Sign Language recognition with various novel approaches to recognize gestures. Also, most of the research is done mainly in American Sign Language (ASL) and not in Indian Sign Language (ISL).

The task is to convert Sign Language (SL) into text or speech. Due to advancements within the field of image processing, an automatic signing converter system is developed. Few researchers have developed tools to assist to convert signing into text or Speech. Researchers within the field of signing are broadly categorized in two ways, Data glove and Image processing. In the data glove system, the user must wear a glove.

In [5] uses a Microsoft Kinect sensor to recognize sign languages. A gesture is perceived as a series of these depth frames generated by the sensor. T. Pryor and colleagues [6] SignAloud is a pair of gloves that uses embedded sensors in the gloves to monitor the location and movement of hands, translating gestures to speech. R. Hait-Campbell et al. [7] created MotionSavvy, a technology that recognizes the hand and arm skeleton using a Windows tablet and the Leap Motion accelerator AXLR8R. Sceptre [8] uses Myo gesture-control armbands to classify signs and gestures using accelerometer, gyroscope, and electromyography (EMG) data. These hardware solutions are accurate, but they are typically costly and inconvenient to transport. By relying on the camera on an Android phone, our device removes the need for external sensors.

The glove consists of a flex sensor, accelerometer, and motion tracker. Sensor output signals are sent to the system for processing and analysis of the gesture that can be then converted into text or speech. In image processing, image is captured through Kinect sensors, leap motion, and web cameras.

The method of acquiring video is either too costly or something one cannot have at all times during a normal conversation. The study of these various systems is done to design a system that can be used in day-to-day life without expensive sensors. Here our solution of developing an android application and using a phone camera for acquiring videos is very easy and simple to use.

III. IMPLEMENTATION: The Whole Cycle

We use an android phone camera to capture the ISL Gestures/ poses shown by a person, so as to decrypt it into English text. This captured ISL frames will be transmitted to a remote server for processing. The frames need to be preprocessed for recognition of gestures. The preprocessing will involve the following steps:

- Stabilization
- Skin segmentation
- Morphological operations to remove background noise

In each case, the person's hand is extracted and tracked. Features from the hand are extracted and fed into a classifier to determine the hand poses. The determined hand pose is then sent to android application in the form of English text. The overview of the application is shown in the figure below:

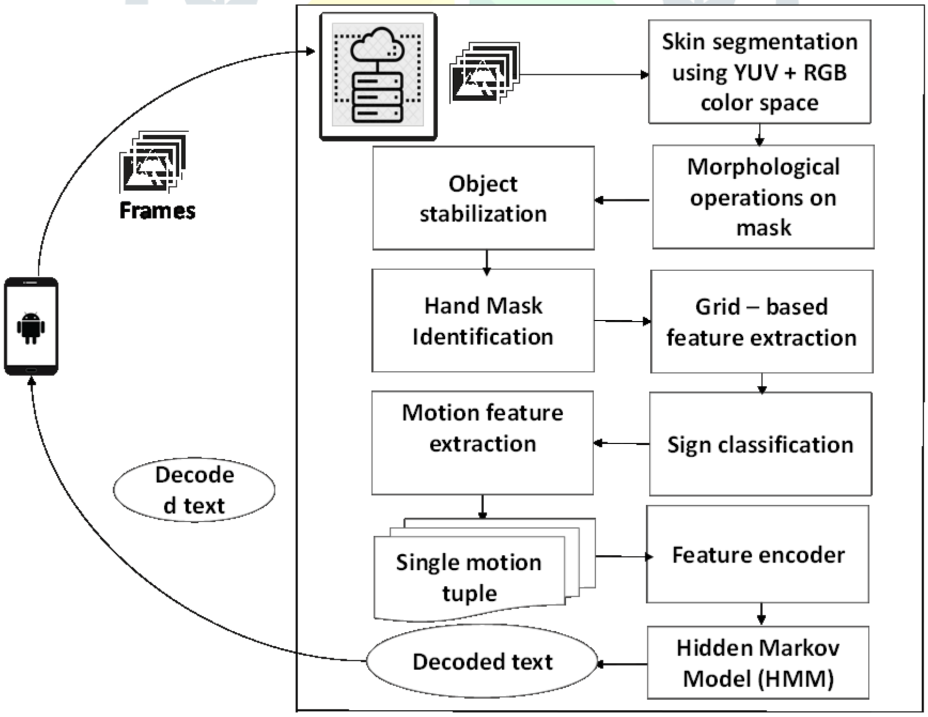
Frames
Skin segmentation using YUV + RGB color space
Morphological operations on mask
Object stabilization
Hand Mask Identification
Grid – based feature extraction
Sign classification
Motion feature extraction
Single motion tuple
Feature encoder
Hidden Markov Model (HMM)
Decoded text
Decode d text

Fig (2): flow diagram for hand poses recognition

A. Dataset:

Around 3000 photos per letter in ISL were recorded. Around 500 images were captured for each of the 6 gesture-related intermediate hand poses. There are a total of 78,500 images in the dataset.

The majority of these photos were taken with a regular webcam, although a few were taken with a smartphone camera. The resolutions of the photos differ. 15 gesture videos were recorded for each of the 12 one-handed pre-selected gestures described in Talking Hands to train HMMs. To make the HMMs more robust, these videos have minor variations in sequences of hand poses and hand motion. The sign demonstrator was wearing a full sleeve shirt in these videos, which have been captured using a smartphone camera.

B. Pre-processing

1. Skin color segmentation

The YUV and RGB dependent skin color segmentation is used to determine skin-like regions in the image, and it yields excellent outcomes. This model was chosen because it produces the best results among the color spaces tested: HSV, YCbCr, RGB, YIQ, YUV, and a few pairs of these color spaces [9]. Using the equation in, the frame is transformed from RGB to YUV color space [15]. The equation is given as (1):

$$\begin{pmatrix} y \\ u \\ v \end{pmatrix} = \begin{pmatrix} +0.299 & +0.587 & +0.114 \\ -0.147 & -0.289 & +0.436 \\ +0.165 & -0.515 & -0.100 \end{pmatrix} \times \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} \qquad (1)$$

The following conditions are used to determine whether pixel is to be considered as skin mentioned in [9]:

$$\begin{cases} 80 < U < 130 \\ 136 < V < 200 \\ V > U \\ R > 80 \text{ and } G > 30 \text{ and } B > 15 \\ |R - G| > 15 \end{cases} \qquad (2)$$

The resultant segmentation mask produces less noise and false positive outcomes. Figure 3 provides an illustration of the segmentation mask.

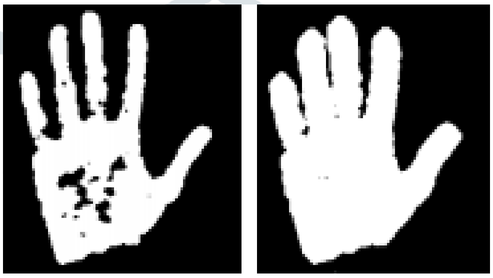

Fig (3): skin segmentation mask (left) and effect of morphology operations (right).

2. Morphological operations

After skin color segmentation, morphology operations were used to eliminate any noise that had been induced. In skin color segmentation, there are two types of errors:

a) Skin pixels are referred to as non-skin.

b) Non-skin pixels are categorized as skin pixels.

There are two basic sub-operations in morphology:

a) Erosion: The mask's active areas (which are white) are diminished in scale.

b) Dilation: In this phase, the mask's active areas (which are white) are enlarged.

The first mistake is tackled by the Morphology Open procedure. It starts with erosion and then advances to dilation. The Morphology Close procedure handles the second mistake. This is attained by dilation and erosion. Figure 3 shows the outcome of implementing morphology operations.

3. Object stabilization

A steady camera position is expected to eventually monitor hand motion. Shaky hands often result in camera movement. False gestures would be identified if the sign demonstrator, that is, the person using ISL, does not shake his hand but the person filming the video shakes his hand. Object stabilization is used to address this problem.

C. Hand extraction and tracking

Hand extraction and monitoring is a vital part of the system since all ISL hand poses and gestures can be interpreted using hand movements. Each frame is pre-processed to produce a black and white image, with white areas representing skin.

We assume that there are no facial regions in these skin zones. In the original illustration, they include sections of the hand and other skin-like parts. Since each frame only includes one hand (the other is hidden) or both hands are touching, the person's hand would be the only prominent contour in the frame.

The centroid of the hand is computed in each frame to track hand motion. The co-ordinates of the centroid of the hand will alter as the hand moves. The slope of the line generated by the centroid of the current frame's hand and the centroid of the previous frame's hand is then defined. The motion of the hand was determined, depending on the value of slope as follows:

- The hand moves leftwards, if -1 < slope < 1 and difference between x co-ordinates of both centroids is positive
- The hand moves rightwards, if -1 < slope < 1 and difference between x co-ordinates of both centroids is negative
- The hand moves upwards, if |slope| > 1 and difference between y co-ordinates of both centroids is positive
- The hand moves downwards, if |slope| > 1 and difference between y co-ordinates of both centroids is negative

An imaginary circle with a 20-pixel radius around the previous hand centroid pixel is known to alleviate noise during hand tracking. If the new centroid is within this range, the change is alluded to as noise, and motion is not taken into consideration. The use of an imaginary circle lowers noise and allows for extremely precise tracking of hand movements.

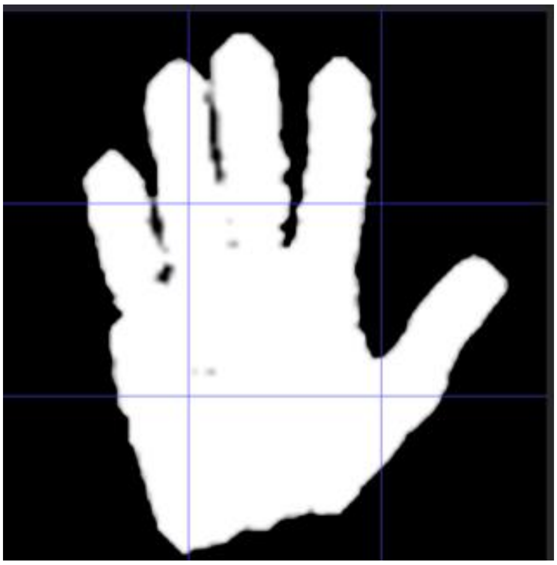

Fig (4): the ISL hand pose fragmented by a 3x3 grid.

D. Feature extraction

A grid-based fragmentation technique is used to extract features. The extracted hand sample is fragmented into M*N blocks using a M x N grid. A feature vector containing M*N feature values is obtained using this grid, with each block offering a feature value. The function value is determined as the percentage of hand contour present in each block. This is stated in the equation (3)

$$\text{Feature value} = \frac{\text{Area of hand contour}}{\text{Area of fragments}} \qquad (3)$$

The function value is 0 if no contour is found. A 3 x 3 grid is built on a sample in Fig. 4 for demonstration purpose. The advantage of this strategy is that the features produced change depending on the orientation of each hand pose. Different hand poses occupy various proportions of grid space and fragment space. As a result, the function vector correctly reflects the hand's shape and location. Each hand pose is interpreted by a different cluster using these M*N features.

Figure 5 backs up these claims. The data in Fig. 5 was analyzed using a 10 x 10 grid, with 100 features extracted per sample. The dimensionality of this data was reduced from 100 to 2 using Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor embedding (t-SNE) [10]. This was accomplished by using PCA to reduce the dimensionality to 40 features, followed by t-SNE to reduce it even further to 2. After visualizing the extracted features, separate clusters depicting different orientations of each hand pose can be seen in Fig. 5.

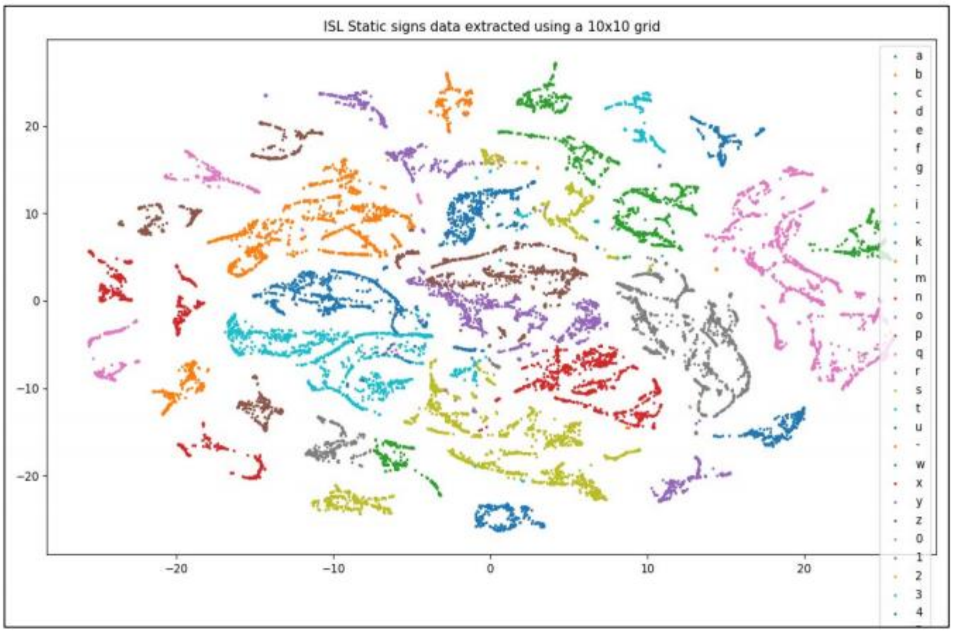

Fig (5): ISL hand poses data visualized using PCA

E. Classification

1. Recognition of hand poses

After looking at the graph in Fig. 5, it is noticeable that the data is organized into clusters. For the same hand posture, there are several clusters. An algorithm capable of efficiently distinguishing clustered data was needed for classification. The K-Nearest Neighbors (k-NN) algorithm was found to be suitable for such data distribution. The k nearest samples, which were previously fitted in the classifier, are computed using Euclidean distance as a distance metric. A brute force approach to distance calculation can be used, in which the Euclidean distance between the sample and each fitted sample in the classifier is measured and the k shortest distances are chosen.

2. Recognition of Gestures

Even when the same sign demonstrator is used, there are still some differences in the signs they make. Some type of statistical model is needed to handle such variations. HMMs (Hidden Markov Models) are a type of statistical model that can effectively manage such variations [12]. Continuous and discrete HMMs are the two kinds of HMMs.

The number of observation symbols that can be used in each element of the observation sequences in continuous HMM is infinite, but it is finite in discrete HMM. The HMM may be left-to-right or ergodic. The change in a left-to-right HMM can only occur in one direction, i.e., if the HMM moves to the next state, it cannot return to the previous state. In ergodic HMM, however, transition from one state to another is possible. For left-to-right HMMs, the initial state probabilities (π) and transformation probabilities are expected.

Gestures are perceived by the human brain as a set of a few intermediate hand poses and hand motions performed in a specific order. ISL Gestures are constituted of intermediate stationery hand poses and the hand movements that bind them. As a result, discrete left-to-right HMM was used in this method. This classifies the given observation sequence as belonging to one of the six predefined movements using the segmented hand centroid motion and pose classification results.

The estimation probabilities and state transfer probabilities of HMM chains are trained after all of the parameters listed here [13] are been initialized. The Baum-Welch algorithm was used [13, 9]. The HMM chains are trained with the help of a gesture database that was developed. The new observation sequence is fed to the HMM chains after they have been educated, and the HMM chain with the highest score using the forward-backward algorithm [13] is calculated to be the recognized gesture.

IV. APPLICATION

The framework is implemented in the form of an Android app. The software uses the camera on the device to record the person's sign language. The frames were taken at a 7 frames per second place. Each frame is sent to a remote server in real time. The processing takes place on the server. The result is sent back to the application, which is shown in the bottom-portion along with the accuracy at which each pose or gesture is classified. AWS api call is currently being used to mimic a client-server relationship. The fig shows the screenshots of application.

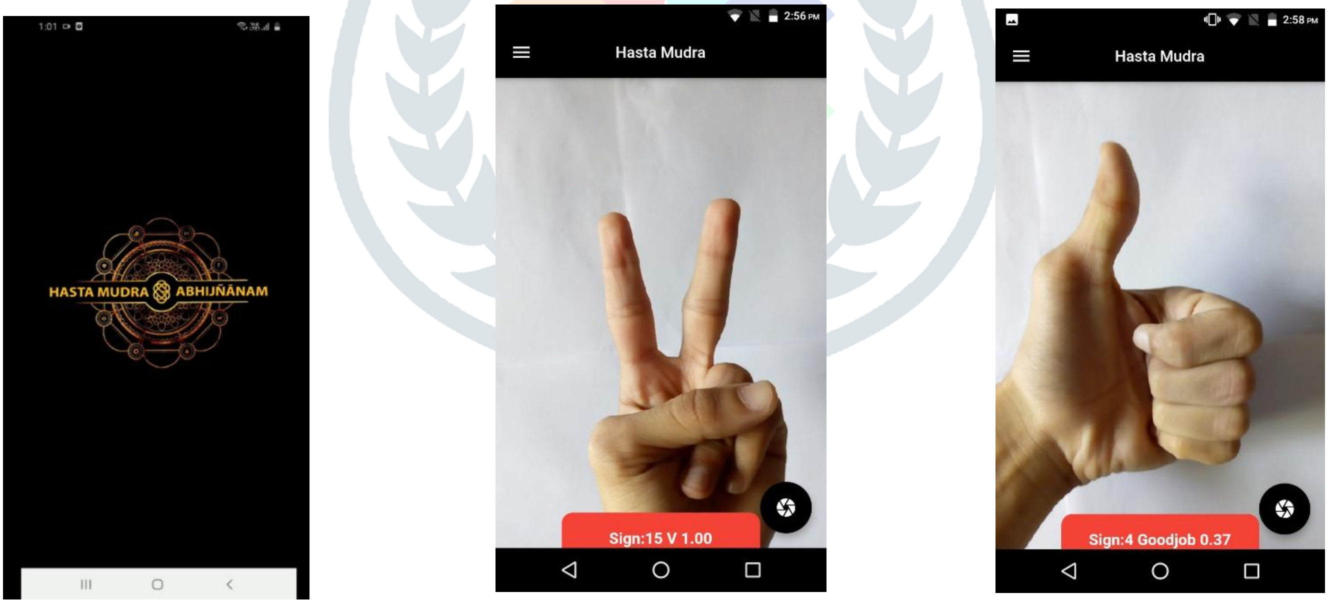

Fig (6): leftmost is home page, center is alphabet recognition (alphabet v), rightmost is gesture recognition (gesture good job)

V. Future scope

Till date, implementation on the basis of 1 hand gestures has been achieved successfully. In the future, we will be working towards gesture capturing and recognition of 2 handed gestures as well.

Currently we have conveniently accomplished the task of converting the hand gestures to English text, with the output being displayed to the user on the device screen that they'll be using. We are planning to work on converting the signs/gestures to English text or speech based of the preferences of the user.

We also intend to introduce a new feature wherein we can import the video gestures or images of hand poses from gallery and interpret the message.

VI. Conclusion

The results show that the method described in this paper can reliably track the sign demonstrator's hand movements using techniques including object stabilization, skin color extraction, and finally hand extraction. It has a 99.7% accuracy rate in classifying all 26 ISL hand poses. With an overall accuracy of 97.23 percent, the machine was also able to distinguish 6 gestures. For each gesture, an HMM chain is used, and a k-NN model is used to distinguish each hand pose. Based on the findings, it can be concluded that the device can accurately and in real-time recognize hand poses and movements in ISL. In terms of sign language understanding, the method is more accurate and quicker than other methods discussed in the literature.

This method is general and can be applied to a variety of one- and two-handed movements. If a dataset that meets the system's current requirements is available, the system outlined in this paper may be applied to other Sign Languages.

VII. Acknowledgment

Mrs.Sushma Khanvilkar directed the creation of this project. Her extensive knowledge of Machine Learning and Computer Vision has given us a greater understanding of the application development process.

References

[1] TalkingHands.co.in, "Talking Hands," 2014. [Online]. Available: http://www.talkinghands.co.in/. [Accessed: 21- Jul- 2017].

[2] K. Shenoy, T. Dastane, V. Rao and D. Vyavaharkar, "Real-time Indian Sign Language (ISL) Recognition," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2018, pp. 1-9, doi: 10.1109/ICCCNT.2018.8493808.

[3] Raheja, J.L., Mishra, A. & Chaudhary, A. Indian sign language recognition using SVM. Pattern Recognit. Image Anal. 26, 434–441 (2016). https://doi.org/10.1134/S1054661816020164.

[4] G. Ananth Rao, P.V.V. Kishore, Selfie video based continuous Indian sign language recognition system, Ain Shams Engineering Journal, Volume 9, Issue 4, 2018, Pages 1929-1939, ISSN 2090-4479, https://doi.org/10.1016/j.asej.2016.10.013.

[5] A. Agarwal and M. K. Thakur, "Sign Language Recognition using Microsoft Kinect," Sixth International Conference on Contemporary Computing (IC3), September 2013.

[6] MailOnline, ''SignAloud gloves translate sign language gestures into spoken English," 2016. [Online]. Available: http://www.dailymail.co.uk/sciencetech/article-3557362/SignAloudgloves-translate-sign-language-movements-spoken-English.html. . [Accessed: 10- Feb- 2018].

[7] Alexia. Tsotsis, "MotionSavvy Is A Tablet App That Understands Sign Language," 2014. [Online]. Available: https://techcrunch.com/2014/06/06/motionsavvy-is-a-tablet-app-thatunderstands-sign-language/. [Accessed: 10 – Feb- 2018].

[8] P. Paudyal, A. Banerjee and S. K. S. Gupta, "SCEPTRE: a Pervasive, Non-Invasive, and ProgrammableGesture Recognition Technology," Proceedings of the 21st International Conference on Intelligent User Interfaces, pp. 282-293, 2016.

[9] Z. H. Al-Tairi, R. W. Rahmat, M.I. Saripan and P.S. Sulaiman, "Skin Segmentation Using YUV and RGB Color Spaces," J Inf Process Syst, vol. 10, no. 2, pp. 283-299, June 2014.

[10] L. Maaten and G. Hinton, "Visualizing Data using t-SNE," Journal of Machine Learning Research, vol. 9, pp. 2579-2605, November 2008.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel et al, "1.6. Nearest Neighbours – scikit-learn 0.19.1 documentation," 2011. [Online]. Available: http://scikitlearn.org/stable/modules/neighbors.html#nearest-neighboralgorithms. [Accessed: 12- Sep- 2017].

[12] C. Vogler, D. Metaxas, "Handshapes and movements: Multiple channel ASL Recognition," Gesture-Based Communication in HumanComputer Interaction, pp. 247-258, 2004.

[13] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of the IEEE, vol. 77, no. 2, February 1989.

[14] L. Baum, "An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Process," Inequalities III: Proceedings of the Third Symposium on Inequalities, ssvol. 3, pp. 1-8, 1972.

[15] B. C. Ennehar, O. Brahim, and T. Hicham, "An appropriate color space to improve human skin detection," INFOCOMP Journal of Computer Science, vol. 9, no. 4, pp. 1-10, 2010.