Sign Language Recognition

Anup Kumar, Karun Thankachan and Mevin M. Dominic
Department of Computer Science and Engineering
National Institute of Technology, Calicut, India 673601

Email: (janupkumar10@gmail.com, thankachankarun639@gmail.com, dominicmevin@gmail.com)

***Abstract*— This paper presents a novel system to aid in communicating with those having vocal and hearing disabilities. It discusses an improved method for sign language recognition and conversion of speech to signs. The algorithm devised is capable of extracting signs from video sequences under minimally cluttered and dynamic background using skin color segmentation. It distinguishes between static and dynamic gestures and extracts the appropriate feature vector. These are classified using Support Vector Machines. Speech recognition is built upon standard module – Sphinx. Experimental results show satisfactory segmentation of signs under diverse backgrounds and relatively high accuracy in gesture and speech recognition.**

***Index Terms*— Computer and information processing, Feature extraction, Gesture recognition, Image processing, Sign Language Recognition, Speech recognition**

I. INTRODUCTION

Sign language is the basic means of communication for those with hearing and vocal disabilities. Those disadvantaged have difficulty in their day to day lives. We aim to develop a system that would ease this difficulty in communication.

Sign language consists of making shapes or movements with your hands with respect to the head or other body parts along with certain facial cues. A recognition system would thus have to identify specifically the head and hand orientation or movements, facial expression and even body pose.

We propose the design for a basic yet extensible system that is able to recognize static and dynamic gestures of American Sign Language, specifically the letters a-z (where j and z are dynamic with hand or fingertip movement while the rest are static). American Sign Language was chosen since it is utilized by a majority of those disabled.

The system is able to perform in dynamic and minimally cluttered background with satisfactory result as it relies on skin color segmentation [3] to separate the gestures, i.e. focus on the hands and face. Since the lexicon under consideration (letters a-z) does not involve facial cues, we eliminate the face using Viola-Jones face detection followed by subtraction of the detected region. We classify the gesture as static or dynamic by measuring the distance moved by the hand in subsequent frames.

For static gestures, we use Zernike moments, a well-known shape descriptor in image processing as shown in [4] [10]. For dynamic gestures we extract a curve feature vector which shows high accuracy in uniquely identifying paths [7].These feature vectors are then classified using pre-trained SVM classifiers. This helps map the gesture to a particular alphabet.

For speech recognition, Sphinx module is used which maps the spoken alphabet to text with high accuracy. This text is then mapped to a picture if it is a static gesture or a video if it is a dynamic gesture.

The major contributions of this paper are as follows –

1) A novel system to aid in communicating with those having speech and vocal disabilities

2) A real-time approach to bare hand detection using face detection and subtraction followed by skin color segmentation with minimal noise and false positives [3] [11].

3) An improvised method to detect center-of-gravity and the fingertips of the hand [11]

4) An improvised method to hand posture [10] and dynamic gesture detection [7]

The paper is organized as follows – Section II describes the past research and advances made in the same fields. In Section III we give a brief overview of our system. Section IV details gesture recognition system step-by-step. Section V briefly discusses the speech recognition system. Section VI shows the experimental results. In Section VII we compare our method with other common and prevalent systems. In the last section, Section VIII we give the conclusion of our method.

II. RELATED WORK

There are two categories for vision-based hand gesture recognition. The 3-D hand model based method and appearance based method. The 3-D hand model based method works by comparing the input frames and the 2-D appearance projected by the 3-D hand model. However a huge database is required to deal with all the possible

978-1-4799-8579-1/16/$31.00 ©2016 IEEE

projections of the 3-D hand model making it less practical. In contrast, appearance-based techniques extract the image features and model these as visual characteristics of a hand posture and compare them with those features extracted from the live video feed of a user performing a gesture. They are real-time in nature because 2-D image features are used. The appearance-based techniques can again be categorized as hand posture detection (static) and gesture detection (dynamic). Nasser H.D et.al [1] considers this approach where the key features extracted are SIFT (Scale Invariant Feature Transform) key-points. They further constructed a grammar from a sequence of hand postures for detecting dynamic gestures.

Emil M.P. et.al [2] proposes the same but uses Haar-like features for describing the image and AdaBoost for classification. It discusses the advantages like being fast in computation and disadvantages such as the need to use a large number of features, which makes the training stage of the system difficult since the AdaBoost classifier is used.

The identification and extraction of hands from a cluttered and dynamic background poses another problem. One of the most popular methods is Skin color detection. In [3] P.K.Bora et.al introduced HSV - Hue, Saturation, Value color space and it states that the skin color, irrespective of gender or race falls in particular ranges for H and S. Utilizing this information we can extract skin colored objects from any background.

In [4] Macheal V et.al had carried out the evaluation of various shape descriptors including Zernike moments and displays its superiority. Zernike moments have become increasingly adapted to image processing schemes as their higher order moments can be calculated independent of lower order moments. Also reconstruction and rotation invariance make them an attractive feature for shape description. Athira et.al in their project report [10] shows the viability of Zernike Moments by building a system to recognize only the static gestures of ASL against a uniform background. It also features a speech engine to convert speech to gesture.

In [5] a basis for usage of Hidden Markov Models (HMM) is established by drawing an analogous relationship between speech recognition and gesture recognition. HMMs can be used to model time series data, and here the movement of the hand along the co-ordinate axis is tracked and each direction is taken as a state. This paper makes use of a lexicon of 40 gestures and achieves an accuracy of 95 percent. It also states its disadvantage that as the lexicon grows the need to describe the hand configuration along with hand trajectory also will grow making the designing of HMM more complex and time-consuming. We needed a method to describe dynamic gesture in a simpler way.

In [6] an object tracking algorithm, Hausdorf object tracker is used to extract frames from a live video feed that will best encompass the translation an object has gone through. This is represented as a motion vector from which a number of static and dynamic feature are extracted for classification.

Emil M.P et.al [7] discusses the extraction of features from the gesture trajectory for dynamic hand gesture recognition. The idea is to track the movement of the COG and extracts a feature vector from the same which consists of parameters like velocity, acceleration etc. To be able to classify the feature vectors, received after pre-processing, a prediction method is required. Classifiers can be trained in order to predict unknown feature vectors. A comparative study in [8] on the performances of various classifiers showed best results with Multiclass SVM. LibSVM, a library for Support Vector Machines is used to implement Multiclass SVM. LibSVM implements it using one-against-one approach. In classification, tournament mechanism is used and the feature vector is assigned to the class that gets maximum votes [9].

III. SYSTEM OVERVIEW

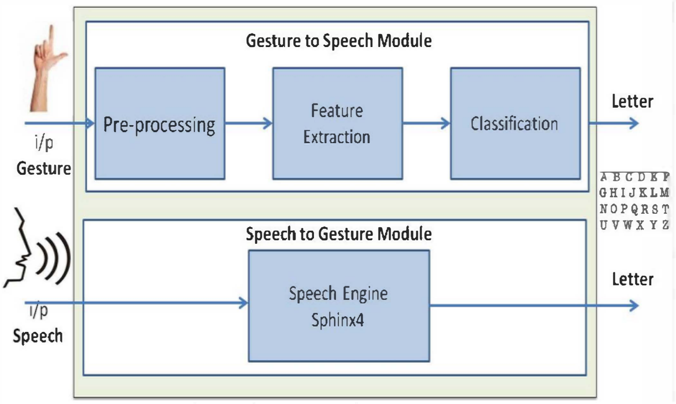

Fig 1: High Level View of System

On a high level the system comprises of two independent modules as shown in the above diagram. The two modules being the 'Gesture To Speech Module' and 'Speech to Gesture Module'. The Gesture to Speech Module takes a gesture as an input in the form of an image/video and produces the speech corresponding to the sign as the output, while the second module does the vice-versa where it takes speech from the user as the input and produces the corresponding image/video as the output. Here the image input/output is meant for a static gesture while a video in the same case is used for a dynamic gesture.

IV. GESTURE RECOGNITION

The input can be a static or dynamic gesture, so to make it as general as possible a video recording of two seconds is passed as the input at the rate of 6fps. The flow diagram of the gesture to speech module is shown below (Fig 2). The gesture is extracted and depending on its type (static/dynamic) certain features are extracted. These are then classified using pre-trained SVM classifiers. The details of image extraction and classification is explained in the below section.

A. Skin – Color Sampling

It is theorized that in the Hue, Saturation, Value color space the skin color, irrespective of gender or race lies in-between certain values of Hue and Saturation[21][22]. However if this global range is used for thresholding it would give us some noise and/or false positives.

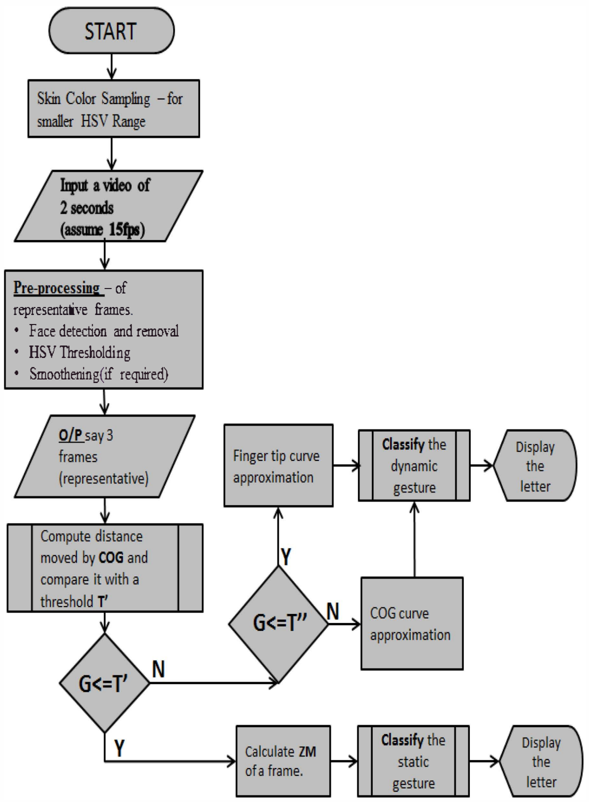
START
Skin Color Sampling – for smaller HSV Range
Input a video of 2 seconds (assume 15fps)
Pre-processing – of representative frames.
• Face detection and removal
• HSV Thresholding
• Smoothening(if required)
O/P say 3 frames (representative)
Compute distance moved by COG and compare it with a threshold T'
G<=T'
N
Y
G<=T"
Y
N
Finger tip curve approximation
Classify the dynamic gesture
Display the letter
COG curve approximation
Calculate ZM of a frame.
Classify the static gesture
Display the letter

Fig 2: Flow Diagram of Gesture Recognition

So we sample [20] the skin color of the user prior to sign recognition as shown in Fig.3. This helps obtain a tighter range for hue and saturation and reduces the noise.

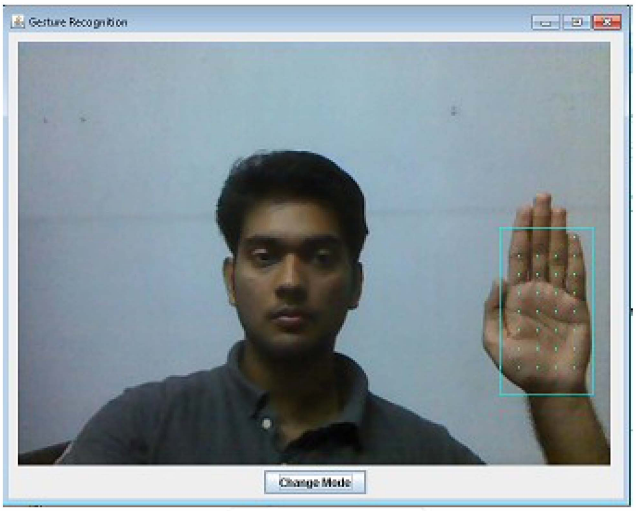

Fig 3: Sampling stage

B. *Image Pre-Processing and Hand Segmentation*

Since skin color segmentation is used to detect the hand we have to make sure that there is no other prominent skin colored regions in the frame. The face is often more prominent than the hand. Hence the first step in image pre-processing is the detection and removal of face using Viola and Jones algorithm. This algorithm uses AdaBoost classifier. The bounding box of the face is obtained and is subtracted from the image, as shown below (Fig.4).

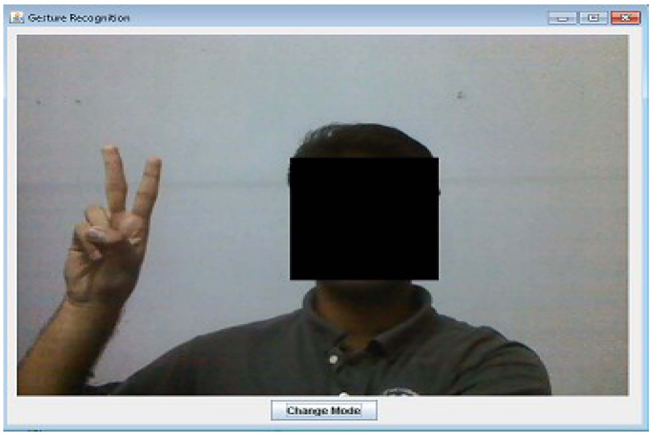

Fig 4: Face Detection and Subtraction

After this we assume that the ASL sign shown by the hand is largest skin colored area in the frame. The hand is isolated by using HSV thresholding using the range sampled earlier. Now we notice that other skin colored regions also gets isolated in the frame which is not of interest (Fig.5).

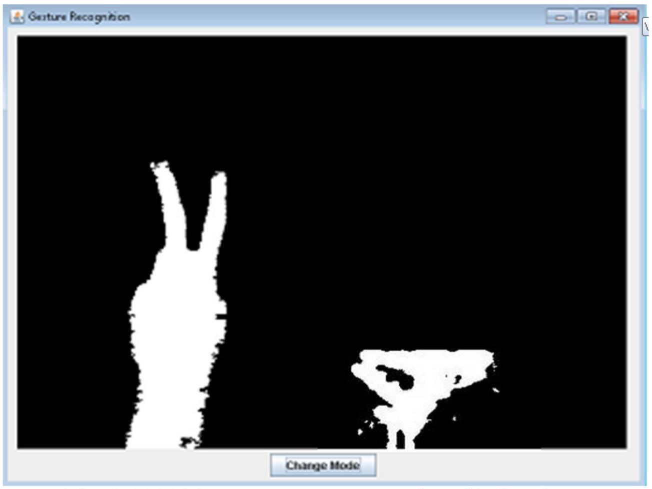

Fig 5: HSV Segmentation

The largest threshold region is extracted to focus in on the hand. Now there are some false negatives (black holes as can be seen in Fig.5) inside the hand and the contour of the hand is disrupted in certain locations. To deal with this we dilate the image, extract the largest contour and fill it. The result is depicted in Fig.6.

C. *Post Processing After Hand Segmentation*

After the image preprocessing, we get a binary image of the hand (Fig 6) which we invert (Fig 7). To be able to recognize dynamic gestures we need to detect and track the finger-tips and COG to know whether they are moving.

978-1-4799-8579-1/16/$31.00 ©2016 IEEE

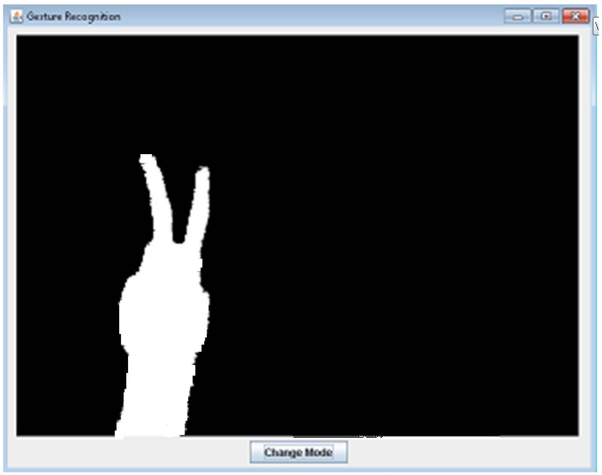

Fig 6: Noise Elimination

This method has been devised by referring several other practices and tailoring it to the current need. The Center of Gravity is computed using standard formulae and is shown by the green dot in the below figure (Fig 7). Now to detect the fingertips, we find all the convexity defects of the hand using its contour (Fig 8.2) and convex hull (Fig 8.3).

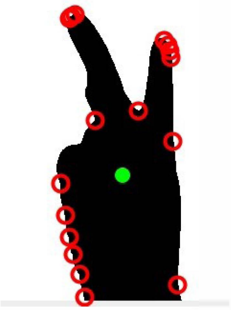

Fig 7: Convexity Defects

These defect points become our candidate finger-tips. To eliminate the non finger-tip points, we devise a three step procedure as described in [11] and [18]. First, among all the candidate points only aculeate points are kept (Fig 8.5). If the angle between a point to the left of a candidate point, the candidate point and a point to the right of it on the contour is < 53 deg, it is considered to be an aculeate point. Fig 8.5 shows the remaining aculeate points.

The points considered to form the angle which is to be tested as aculeate or not were taken at distance of 50 points along the contour. Angle 53 was determined after experimenting with many values.

Second, among all the aculeate points that are left there may be a cluster of points especially at the actual finger tips as can be seen in Fig.8.5. To eliminate these clusters for all candidate points having other candidate points in its vicinity i.e. within 50 points on the contour, a single representative point is kept (Fig 8.6). The only non finger-tip points that remain after applying 50 point approximation are the aculeate points between fingers.

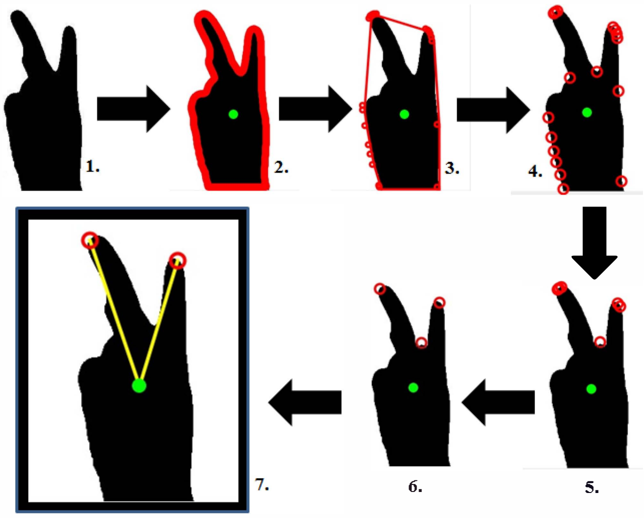

Fig 8: Finger-tip detection

Third step is to distinguish between actual finger-tips and aculeate defect points between fingers. The angle between COG, the candidate point and a point (distance of 50 points along the contour) to the right of candidate point is calculated. This angle is always >90 degree for aculeate points between fingers and thus can be eliminated (Fig 8.7). So now we have detected the actual fingertips.

D. Threshold Comparison

Now that we have detected the CoG and fingertips we can identify if the gesture is static or dynamic by tracking the distance they move. In this paper the entire class of ASL gestures has be divided into three. First there is the basic classification of Static gestures and Dynamic gesture. Dynamic gestures are then further subdivided into dynamic gestures with significant hand movement and dynamic gestures with significant fingertip movement. Significant movement of the hand means substantial movement of the CoG of the hand while in the other class CoG almost remains stationary but the fingertip moves substantially.

To identify the category the gesture belongs to we have two decision nodes (Fig.2). We compute the distance moved by the Center of Gravity (CoG) of the hand in subsequent frames, let it be 'G' as represented in the flow diagram. This distance G is compared with at-most two thresholds, T' and T'', at two separate stages of the flow (Fig.2). In the first stage if G is less than the first threshold T' we assume that the hand is stationary and classify it as a static gesture and if it is greater there is some significant movement of the CoG and thus classify it as a dynamic gesture. This decision node is a compulsory decision node, and the second comparison with T'' is carried out only in the case of dynamic gestures. If G is greater than T" it is a dynamic gesture with significant hand movement and if lesser it is a dynamic gesture with significant fingertip movement. In both cases of dynamic gestures we track the CoG or the fingertip as the case maybe and approximate the curve drawn.

978-1-4799-8579-1/16/$31.00 ©2016 IEEE

E. Feature Extraction

For static gestures, we use Zernike moments to identify the orientation of the hand. From experimental results and [10] we were able to deduce Zernike moments up to the seventh order and hence a seven feature vector is sufficient to identify a static gesture.

For dynamic gesture involving the tracking of CoG or a particular fingertip we characterize the path traced by a curve feature vector of 5 elements. To do this we first identify Key points or representative points on the path. As demonstrated below a circle would require more key points to characterize the path whereas a square would only require four.

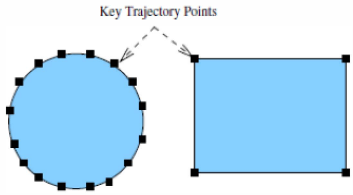

Fig 9: Key point for circle and rectangle [6]

Next we extract the features which are

1) Trajectory Length/Location feature:

The total length of the trajectory can be calculated by computing the distance between the key-points and summing them up. It is translation invariant as even if there is a shift the location where the gesture is drawn the length of the gesture will not change.

Location feature is the distance between the center of gravity of the trajectory and the key-points. Since there are a number of key points the average of the sum is considered. It is also translation invariant as distance between center of gravity and key points would not vary even if there is a shift in location of the gesture.

Thus the ratio, *(Trajectory length/Location feature)* would be a scale and translation invariant ratio.

2) Number of significant changes in hand orientation:

The orientation change (i.e. the angle formed) along the set of three key-points is considered, if it is greater than a predefined threshold say, an angular displacement >= 45 it is counted as one significant change, else ignored. The total count of such changes is accounted here.

3) *Average velocity*: It is the average velocity over the whole trajectory.

4) *Standard deviation of the speed:* Variation of the speed over the whole trajectory length indicates the smoothness of the trajectory.

5*) Number of minima velocity points:* Number of points that have a speed below some threshold T-v(min).This indicates turns in the trajectory and hence the smoothness of the trajectory.

F. Classification

Sanjay in his master's thesis [16] and Neelam et.al in their review [17] show that SVM classifiers are a good choice for classification of hand gestures, with some variation for different methods adopted for feature extraction. SVM classifiers are trained and a classification model is built using Java Machine Learning library- JAVAML [19]. Two classification models, one for Zernike feature vectors and another for trajectory feature vectors, are built which successfully classified the gesture to corresponding alphabet.

V. SPEECH RECOGNITION

Speech Recognition (SR) is the conversion of spoken words into text, also known as "Automatic Speech Recognition" (ASR). There are two types of SR Systems, Speaker-independent [12] and Speaker-dependent systems, where the latter has a training phase which uses a person's specific voice for the recognition purpose. A number of Speech applications are available in the market like Dragon NS [13]. For experimental purposes of this paper the Speech recognition module was setup in two ways, first using a speech engine (Sphinx 4), where a restricted set of grammar for alphabets from A-Z was used, just like a command recognition system. The SPHINX Speech Recognition Engine of CMU [14] was utilized. A SPHINX engine just like any other Speech Engine requires two files as input an Acoustic Model and a Language model or a Grammar File. This general model of SPHINX is depicted in below diagram (Fig 9). Its working is based on Hidden Markov Models (HMM).

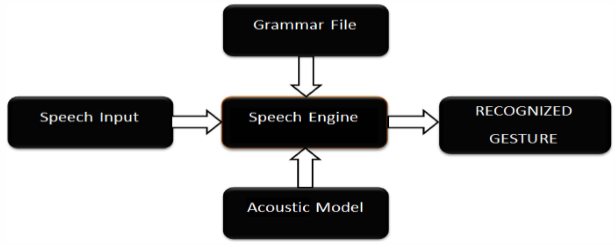

Fig 10: General Model - Speech Engine

Second, it was implemented in android considering the ease with which a normal user can use it to communicate with someone using ASL. The 'SpeechRecognizer' Class [15] of android is used for the same. The interface of the application when letter 'b' is given as the input is shown below (Fig 10). The flow of the Android Application comprises of a decision node which decides whether the speech input corresponds to a letter involving a static or a dynamic gesture for which an image is displayed or a video is played respectively.

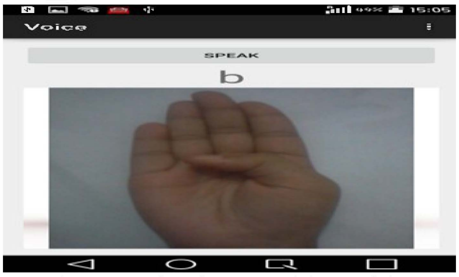

Fig 11: Interface for a Speech input of 'b'

The flow is shown in the below diagram (Fig 11).

978-1-4799-8579-1/16/$31.00 ©2016 IEEE

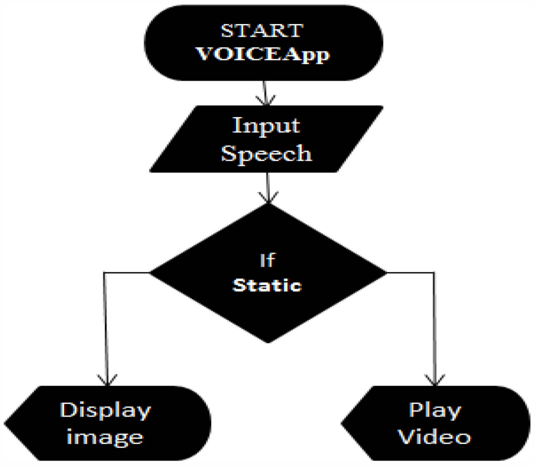

Fig 12: Flow

The SPEAK button (Fig.11) can be used to give a speech input to the system and the ouput is either an image or a video corresponding to the gesture depending on whther it is static or dynamic. The accuracy shows huge variation based on the type of usage and the amount of background noise.

VI. EXPERIMENTAL RESULTS

HSV Segmentation and *Finger-tip detection* showed satisfactory results in constrained environment, i.e. proper lighting and background with limited skin-colored objects. As an example the result of the process applied for alphabets Z and J, are shown:

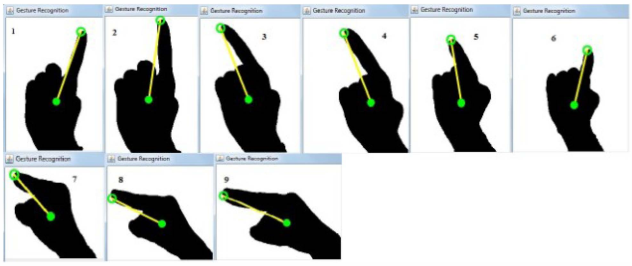

Fig 13: Finger-tip detection for 'Z'

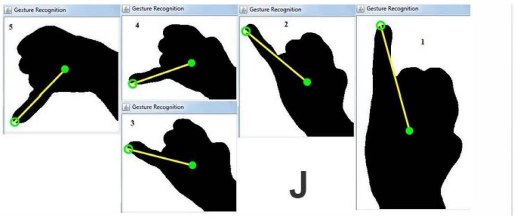

Fig 14: Finger-tip detection for 'J'

Static Gesture recognition was carried out on a lexicon of 24 alphabets (a-y, excluding j) and it succeeded with approximately **93% accuracy** . Results for 'A' and 'C' are shown below:

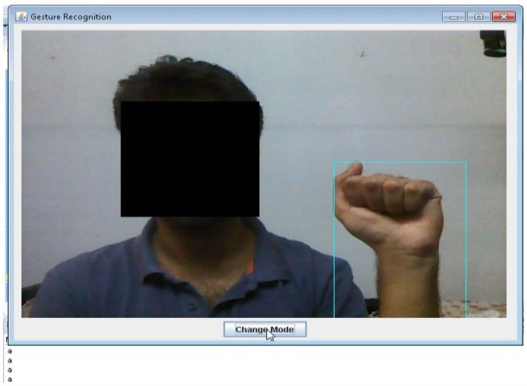

Fig 15: Gesture recognition for 'A'

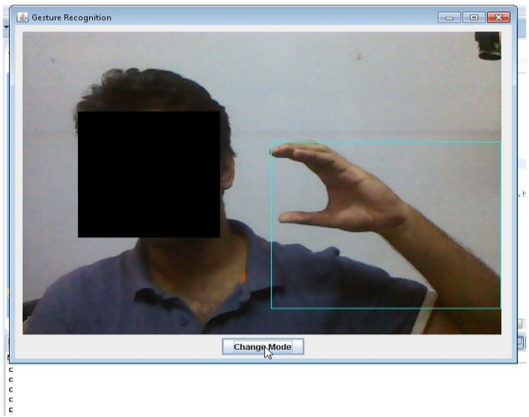

Fig 16: Gesture recognition for 'C'

Dynamic gesture recognition was conducted on a lexicon of 4 gestures (j, z, no, bye) and an **accuracy of 100%** was achieved

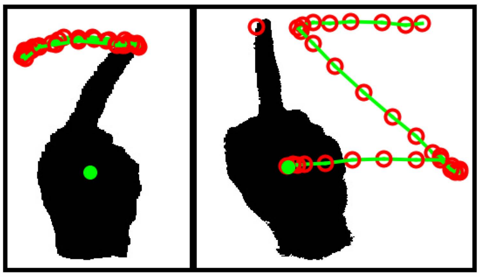

Fig 17: Curve traced for 'no' and 'z'

VII. COMPARISON WITH OTHER METHODS

In the proposed method, HSV is used for segmentation, Zernike moments and curve features for feature extraction and multiclass SVM for classification.

There are a number of other possible methods. Based on how we receive input, we have sensor, marker and data glove methods. Depending on how we constraint the environment i.e amount of lighting, whether a uniform and static background we have different approaches. For segmentation of the sign we can use YCrCb color space, use an active shape model, or use the minimizing of kinetic energy approach. The common feature that can be extracted are SIFT or HAAR features and the classifiers that can be used are Adaboost, HMM and fuzzy-k-means etc. In [11]

978-1-4799-8579-1/16/$31.00 ©2016 IEEE

there is a detailed analysis of a number of common methods and its comparative study is tabulated below (Table 1). The entry (7) indicates our results.

Table 1: Comparison with popular methods [11]

Sl. No.	**Background**	**Segmentation Technique**	**Feature Vector Representation**	**Classifier**	**Acc (%)**
1	Data Glove	Discontinuity (time variant parameter detection)	Posture, position, orientation and motion	HMM	80.4
2	Cluttered	Manually from Background	Blob and ridge features	Particle filtering	86.5
3	Uniform	-	State and transition features	HMM	85
4	Uniform	HSV color space	Convex hull of the contour	Haar like Technique	N/A
5	Uniform	RGB to Gray	Euclidean Distance	Neural Network	N/A
6	Cluttered	HSV color space	SIFT features	Multiclass SVM	96.25
7	Minimally Cluttered	HSV Color Space	Zernike Moment and Curve Feature	Multiclass SVM	>90

VIII. CONLCUSION

The system is novel approach to ease the difficulty in communicating with those having speech and vocal disabilities. Since it follows an image-based approach it can be launched as an application in any minimal system and hence has near zero-cost.

There are further areas of improvement such as increasing the system performance under robust and unfavorable environment (lot of clutter, poor lighting). We also need to expand the current feature set to be able recognize more gesture (like those involving two hands or facial cues). We also need to deal with co-articulation.

REFERENCES

[1] Nasser H. Dardas and Nicolas D. Georganas. Real-time handGesture detection and recognition using bag-of-features and support vector machine techniques. IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, 2011.

[2] Emil M. Petriu Qing Chen, Nicolas D. Georganas. Real-time vision-based hand gesture recognition using haar-like features, 2007.

[3] P.K. Bora M.K. Bhuyan and D. Ghosh. Trajectory guided recognition of hand gestures having only global motions. International Science Index, 2008.

[4] Michael Vorobyov. Shape classi_cation using zernike moments, 2011.

[5] Thad Eugene Starner. Visual recognition of american sign language using hidden markov models. Master's thesis, Massachusetts Institute of Technology, Cambridge MA, 1995.

[6] P.K. Bora M.K. Bhuyan and D. Ghosh. Trajectory guided recognition of hand gestures having only global motions. International Science Index, 2008.

[7] Emil M. Petriu Qing Chen, Nicolas D. Georganas. Feature extraction from 2d gesture trajectory in dynamic hand gesture recognition, 2006.

[8] Manisha M. Ingle Neelam K. Gilorkar. A review on feature extraction for indian and american sign language, 2014.

[9] Chih-Chung Chang and Chih-Jen Lin. A review on feature extraction for indian and american sign language, 2013.

[10] ATHIRA P K ALEENA K RAJ and DEEPA I K. Sign language conversion software for people with hearing and vocal disabilities,2013.

[11] Real-Time Palm Tracking and Hand Gesture Estimation Based on Fore-Arm Contour, 2011

[12] Fifth Generation Computer Corporation-"Speaker Independent Connected Speech Recognition" [Online]-Available: Fifthgen.com

[13] NUANCE.Dragon Speech Recognition Software [Online]. Availabe: http://www.dragonsys.com

[14] Kai-Fu Lee, Hsiao-Wuen Hon, and Raj Reddy, An Overview of the SPHINX Speech Recognition System. IEEE Transactions on Acoustics, Speech and Signal Processing

[15] Developer-Android – "SpeechRecognizer" [Online] – Available: developer.android.com

[16] SANJAY MEENA. A Study on Hand Gesture Recognition Technique, 2011.

[17] Neelam K. Gilorkar, Manisha M. Ingle. A Review on Feature Extraction for Indian and American Sign Language, 2014.

[18] Java Advent Calendar, Hand and finger detection using JavaCV [Online]. Available: http://www.javaadvent.com/2012/12/hand-and-finger-detection-using-javacv.html

[19] Java-ML, Java Machine Learning Library [Online]. Available: http://java-ml.sourceforge.net

[20] Hand Tracking and Recognition With OpenCV [Online]. Available http://simena86.github.io/blog/2013/08/12/hand-tracking-and-recognition-with-opencv/

[21] Mokhtar M. Hasan & Pramod K. Misra, HSV Brightness Factor Matching for Gesture Recognition System. International Journal of Image Processing (IJIP), Volume (4): Issue (5)

[22] Xingyan. Li. "Vision Based Gesture Recognition System with High Accuracy". Department ofComputer Science, The University of Tennessee, Knoxville, TN 37996-3450, 2005

[23] Dealing with Noise [Online]. Available: http://stackoverflow.com/questions/22898996/how-to-remove-unwanted-lines-noise-in-opencv