American sign language translation using edge detection and cross correlation

Conference Paper · August 2017

DOI: 10.1109/ColComCon.2017.8088212

CITATIONS
6

READS
215

3 authors:

Anshal Joshi
University of Puerto Rico at Mayagüez
1 PUBLICATION **6** CITATIONS
SEE PROFILE

Heidy Sierra
University of Puerto Rico at Mayagüez
50 PUBLICATIONS **265** CITATIONS
SEE PROFILE

Emmanuel Arzuaga
University of Puerto Rico at Mayagüez
29 PUBLICATIONS **376** CITATIONS
SEE PROFILE

American Sign Language Translation Using Edge Detection and Cross Co-relation

Anshal Joshi, Heidy Sierra and Emmanuel Arzuaga

Department of Electrical and Computer Engineering
University of Puerto Rico Mayaguez, Puerto Rico

Keywords: American Sign Language, Computer Vision, Classification, Edge Detection.

Abstract

According to the World Health Organization (WHO), there are approximately 360 million people worldwide that have disabling hearing loss and 70 million that are mute. Developing advance communication for them is very complex and its been a difficult task for many years. Currently, American Sign Language, which is expressed through the hands and face and perceived through the eyes, is the standard language of communication for the Deaf community. Our main focus here is to implement an automated translation system which can translate the American Sign Language to English text using common computing environments such as a computer and a generic webcam. In this investigation, a real-time approach for hand gesture recognition system is presented. Two different approaches are used to translate English letters and words. In the method to recognize letters, first, the hand gesture is extracted from the main image by the image segmentation, morphological operation and edge detection technique and then processed to feature extraction stage. And for the words, a video sequence is captured then divided into frames and process them for the frame selection stage. In frame selection stage, frames are sampled and selected for feature extraction and then the gesture is extracted from all of the frames by the same using the same technique as image segmentation, morphological operation, edge detection technique and combined by Montage. In feature extraction stage the Cross-correlation coefficient is applied on the gesture to recognize it. In the result part, the proposed approach is applied on American Sign Language (ASL) database.

1 Introduction

Communication and community are significant parts of human life. Deaf people are isolated from the most common forms of communication in today's society such as warnings and sound alerts, or any other form of oral communication between people in regular daily activities like visiting the doctor or communicating in the street. In other words, deaf people can often feel disassociated and thus find it hard to get information or help in daily activities or even encountered in emergency situations. As a consequence, deaf people are twice as likely as hearing people to be affected by depression, anxiety and similar problems.

A deaf person mostly relies on vision for clues to what people are communicating as well as other clues like vibrations, sense of touch in floors or around them. Often other people will change the way they act towards deaf people and often become irritated with having to repeat statements, or even become frustrated that they cannot understand each other.

This paper is focused on the implementation of an efficient translation system which can identify all the American Sign Language (ASL) alphabets and convert them into English alphabets. We are creating an ASL database of signs with different hand samples to use as training data for our algorithms. In order to create the sign database we have implemented Graphical User Interface (GUI) Application to capture signs. The GUI will capture the image within certain boundaries, and then image segmentation and morphological filtering algorithms are applied to reduce noise and convert the captured image into binary form. As few signs are very much similar to each other and very hard to differentiate in a black and white image, we use an edge detection technique to include the edges of the fingers as new features that can help us better to discriminate patterns in the gesture. After adding these edges, the training image is saved in sign database that will be later used for the classification of real time gestures.

For the translation, we have implemented a second GUI in which we have a video frame where the camera output can be observed. This application is designed to collect video frames from the camera and extract the sign from an image frame to compare it to the image database and check which of the signs best matches the extracted sign. Finally, the application generates a display of the classified frame, which is an English translation of the ASL sign. In our initial attempt we have implemented a generic sign translator which anyone can use at their home very easily using their everyday devices like a laptop computer. The clear advantage of such approach is that the users do not have to spend extra money to buy additional or expensive hardware.

2 Previous Work

Researches in china created a sign language translator using the Microsoft kinect motion sensing camera [1]. Kinect has been designed for Xbox gamming, in which it can read a specific

movement of the human body and translate them into game control commands using its special sensors. In November 2010 after the release of Microsoft Kinect, a number of researchers almost immediately focus their interest in this area. In 2011, Microsoft released the SDK for kinect, which gave a boost to all the interested researchers. At Microsoft Research Asia, head researcher Ming Zhou proposed their work in sign language translation. They have been able to create a translation system that can capture sign convert them into written text and spoken translation in real time. The non signer is represented by an avatar which takes his spoken words and then accurately converts them into written text so that the deaf person can read it.

In April 2016, two researchers from University of Washington won the Lemelson-MIT student prize $10,000 for the development of gloves that can translate ASL signs into speech [2]. Their invention is called SignAloud, in which each glove contains sensors that captures the hand position and motion and send it to computer via Bluetooth, then the computer searches for the appropriate hand gesture through various sequential statistical regressions, similar to a neural network. If the data match a gesture, then the associated word or phrase is spoken through a speaker.

Kulkarni [8] recognize static posture of American Sign Language using neural networks algorithm. The input image are converted into HSV color model, resized into 80x64 and some image preprocessing operations are applied to segment the hand [8]from a uniform background, features are extracted using histogram technique and Hough algorithm. Feed forward Neural Networks with three layers are used for gesture classification. 8 samples are used for each 26 characters in sign language, for each gesture, 5 samples are used for training and 3samples for testing, the system achieved 92.78 % recognition rate using MATLAB language.

Hasan [6] applied scaled normalization for gesture recognition based on brightness factor matching. The input image with is segmented using thresholding technique where the background is black. Any segmented image is normalized (trimmed), and the center mass [6] of the image are determined, so that the coordinates are shifted to match the centroid of the hand object at the origin of the X and Y axis. Since this method depends on the center mass of the object, the generated images have different sizes, for this reason a scaled normalization operation are applied to overcome this problem which maintain image dimensions and the time as well, where each block of the four blocks are scaling with a factor that is different from other blocks factors. Two methods are used for extraction the features; firstly by using the edge images, and secondly by using normalized features where only the brightness values of pixels are calculated and other black pixels are neglected to reduce the length of the feature vector. The database consists of 6 different gestures, 10 samples per gesture are used, 5 samples for training and 5 samples for testing. The recognition rate for the normalized feature problem achieved better performance than the normal feature method, 95% recognition rate for the former method and 84% for the latter one.

In recent years, with the introduction of a new approach [9] (which has a high detection rate), new studies are mostly concentrated on Boosting and HMM. The most tempting side of using those methods is that they usually work with grayscale images instead of colored images and thus it eliminates the drawbacks of such color based noise issues. This innovative approach is using a well-known technique namely Adaboost classifier which was mentioned in [3].

3 ASL Sign Database Creation

ASL Sign Database is a collection of different gestures used in American Sign Language. It consist of ASL letters (A to Z) and ASL words (like Good Morning, Hi, Bye). The purpose of ASL sign Database is to provide a data training set to our algorithm. We have created two systems to generate the database for English alphabets and words. Which are explained as follows:

3.1 Alphabet Database

In order to create the ASL sign database for alphabets, we have implemented a sign image acquisition software. This application provides a graphical user interface (GUI) environment that allows the user to collect the ASL image sings. These sign images will be used to train our system. see Figure 1.

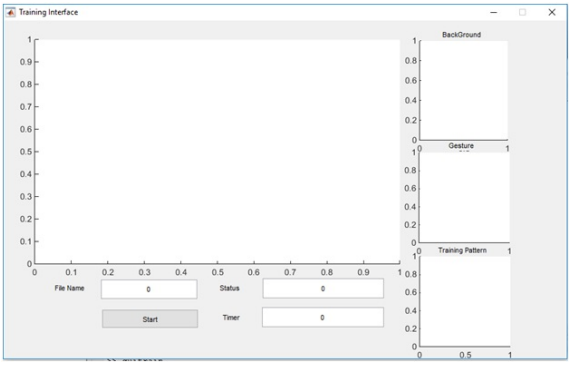

Figure 1. Training GUI.

To start collecting samples live, a user needs to write the file name in the File Name field and press start. when start button is clicked the training function is called and it initiates the camera view. During the training process the system captures an image of the background. Once the background is captured, system will move to capturing gestures. Once both background and gesture are captured, the camera view will stop. The system will start the image segmentation. It is a process in which we convert a RGB image or gray scale image into binary (Black and White) image. This simplifies our classification algorithm to discriminate two objects i.e. black (background) and white (hand) in our image. To obtain best result we have to choose best possible threshold value and segmentation can be done according to that value. Otsu algorithm [7] is used to convert image into binary [5]. Suppose there are two classes of pixels with a0 as background pixel and b1 as and pixel. a0 shows the pixels with intensity level [1,2.....K] and b1 shows the pixels

with intensity level [K+1.....L], from these classes we get the threshold value K* which is in between value of K and K+1 and now hand pixel is assigned value 1 and background pixel is assigned value 0 and we get our desired binary image see Figure 2 and Figure 3

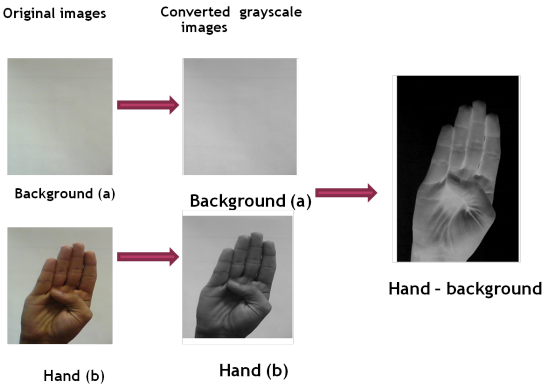

Figure 2. Image Segmentation part 1.

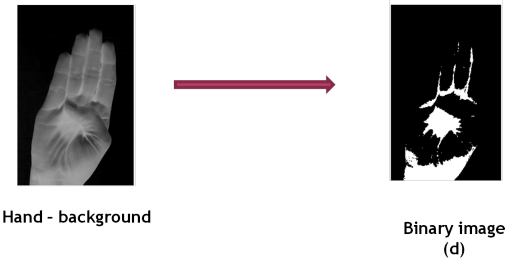

Figure 3. Image Segmentation part 2.

The segmented images we obtain after applying the Otsu algorithm still needs more processing to remove unwanted data and errors. For example there might still remain some background parts containing 1s and some hand parts which denote 0s. In order to remove that noise we have to apply morphological filtering techniques on those segmented images. Dilation' Erosion' Opening' and Closing is the basic operator that work in morphological filtering [7]. Figure 4.

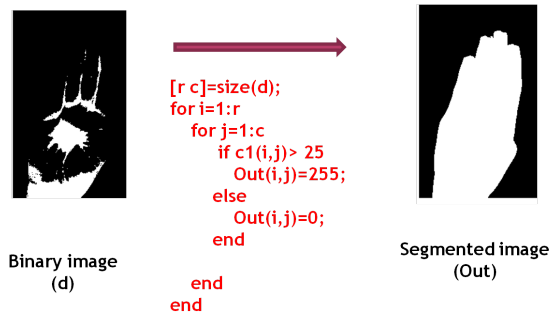

Figure 4. Morphological Filtering.

Now we have the morphed image for the sign but its a binary image with very less details and in binary form few of the signs look similar to each other and thus they very difficult to identify. We apply edges of the original image using edge detection algorithm [5]. Figure 5 depicts this process.

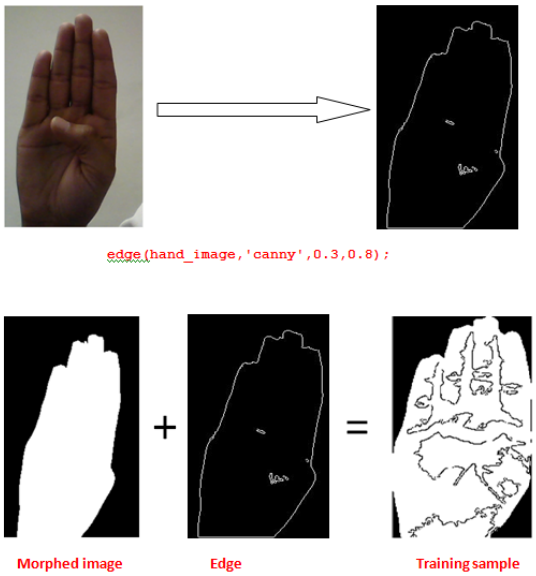

Figure 5. Edge Detection and Final Training Sample.

For each alphabet there are 10 different samples.

3.2 Words Database

In order to create words database, we first need to know that words in sign language consist of several hand movements. That means a word consists of more then one gesture in it. Therefore we need a sequence of images that can accumulately describe an English word. In this work we use a frame selection method that captures sequence images. In this method the frame rate is 35 frame/sec with a sampling factor of 4. The number of frames (nFrames) is calculated by :
nFrames = floor(NumberOfFrameInVideo / sampling_factor);

Then we apply the frame selection method to select the frames from the video we captured from live cam for further processing.
for i = 1 :nFrames
IMG = read(VideoObj, (k-1)*sampling_factor+1);
end

After all the frames are selected, each frame passes through the image segmentation, morphological filtering and edge detection phase mentioned in the above section. Then our system selects all the images and display them using montage. Montage(I) displays all the frames of a multiframe image array I in a single image object. I can be a sequence of binary, grayscale, or truecolor images. A binary or grayscale image sequence must be an M-by-N-by-1-by-K array. A truecolor image sequence must be an M-by-N-by-3-by-K array. This montage image is

our words database sample for a word. see Figure 6

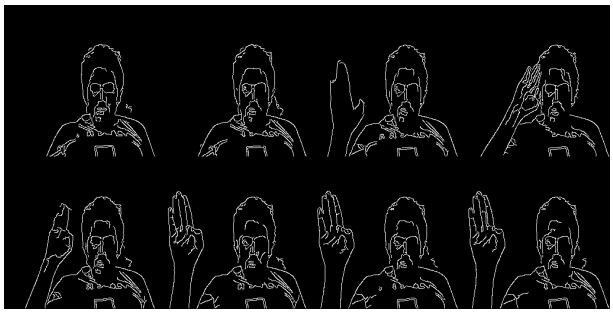

Figure 6. Words Database Image Sample.

4 Sign Translation

Sign Translation begins with the extraction of feature for gesture recognition. Feature extraction and matching is performed using the image Cross-correlation Coefficient. In signal processing, cross-correlation is a measure of similarity of two waveforms as a function of a time-lag applied to one of them. We use this function for matching of hand gesture. Cross correlation is usually applied to find the offset between two similar but time-shifted functions.If a and b are two discrete-time sequences,Cross-correlation measures the similarity between a and shifted (lagged) copies of b as a function of the lag. If a and b have different lengths, the function appends zeros at the end of the shorter vector so it has the same length, N, as the other. The cross correlation coefficient is defined in Equation: 1.

$$\gamma(x,y) = \frac{\sum_s \sum_t \delta_{(x+s,y+t)\delta} \delta_T(s,t)}{\sum_s \sum_t \delta^2_{(x+s,y+t)\delta^2} \delta_T(s,t)} \tag{1}$$

Where

$$\delta_{(x+s,y+t)=I(x+s,y+t)-I'(x,y)}$$

$$\delta_T(s,t) = T(s,t) - T',$$

$$s \in \{1,2,3,...,p\},$$

$$t \in \{1,2,3,...,q\},$$

$$x \in \{1,2,3,..,m-n+1\},$$

$$y \in \{1,2,3,...,n-q+1\},$$

$$I'(x,y) = \frac{1}{pq} \sum_s \sum_t I(x+s,y_t)$$

$$I' = \frac{1}{pq} \sum_s \sum_t T(s,t)$$

The value of cross-correlation coefficient γ ranges from [-1 to +1] corresponds completely not matched and completely matched respectively. For template matching the template, T slides over I and gamma is calculated for each coordinate (x, y). After calculation, the point which exhibits maximum gamma is referred to as the match point. The following step is used for matching of hand gesture:
Step 1: A hand gesture template of size m n is taken.
Step 2: The normalized 2-D auto-correlation of hand gesture template is found out.
Step 3: The normalized 2-D cross-correlation of hand gesture template with various template is calculated.
Step 4: The mean squared error (MSE) of auto correlation and cross-correlation of different sample are found out. The minimum MSE is found out and stored.
Step 5: The corresponding minimum MSE represent the recognized gesture.

We have a GUI which consist of two different mechanism to translate alphabets and the words. see Figure 7

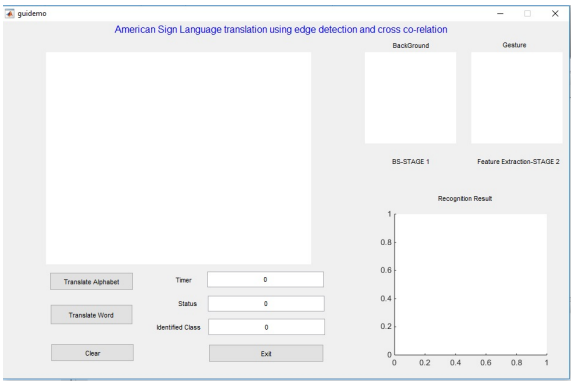

Figure 7. Translation GUI.

4.1 Alphabet Translation

Translate Alphabet button on translation GUI open a camera view in the window. It will follow the same steps of capturing background and gesture and then do image segmentation, morphological filtering and edge detection as explained in Section 3.1.1. The processed image is now compared with the alphabet database using the cross-correlation explained in Section 3.2. After this comparison the result will be displayed in the image box named Recognition Result and the alphabet in the text box named Identified Class see Figure 7.

4.2 Word Translation

The Translate word button on Figure 7, also opens a camera view in the application window and then capture and select the image sequences. Each frame is processed through image segmentation, morphological filtering and edge detection. After processing the frame sequence, these processed frames converted into montage as mentioned in Section 3.1.2. At the end the montage image is compared with the database and the resultant image is displayed in the image box Recognition Result and the word is displayed in Identified Class in the translation GUI.

5 Motionsavvy Platform

Motionsavvy [4] is a company that provides communication solutions between businesses and deaf and hard of hearing customers with recent developments in gesture recognition and machine learning technology. Since 2011, Motionsavvy has been building a database of signs that can be used by various industries with communication needs. Motionsavvy created a tablet Uni, which is designed for two way communication between the deaf and the hearing using two distinct technologies. Using the integrated camera and a special software, the gadget recognize the sign and translated in spoken English and when a hearing person respond by speaking back, the gadget translate the voice into text and display it on the screen. Our system is comparable to motionsavvy, they also provide a training program for user where user can train the system according to them like our system. They also provide a minimal dataset for ASL. The difference is they use a dedicated hardware Uni but our system can be run in any computer. Motionsavvy has sign to voice conversion but in our system only sign to text conversion is there voice conversion is proposed as future work.

6 Results

Classification accuracy for ASL translation is presented. Different test data image is compared against each sample data images several times in order to find the accuracy of the system. Each English letter has 10 different sample images and each word has a sample montage (which is a sequence of image frames stored in database).

Our suggestive method have been done on Intel Core i3-2330M CPU, 2.20 GHz with 4 GB RAM under Matlab R2015a environment. Figure 8 shows the face of worked systems.

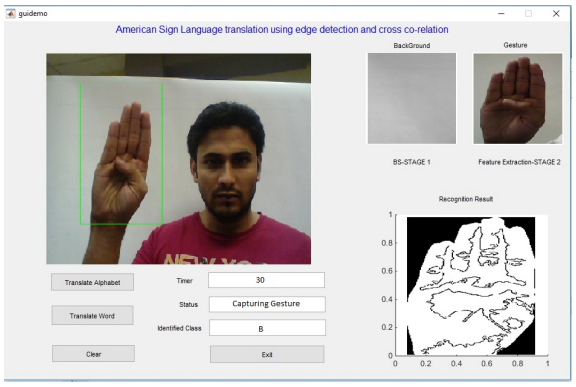

Figure 8. ASL Sign Language Translation System.

In this study, for experimental analysis, we had applied the above mention technique on our database of American Sign Language which consists of 260 images i.e. 10 images per character and we was able to recognize 26 characters out of 26 characters from sign language. Figure 9 shows the accuracy rate for each hand gesture.

In our experiment (with the 94.23% accuracy for alphabets), we observed confusion hand gesture in the recognition phase between some signs. The major confusions were

English Alphabet	Database Images	Recognised Images	Performance
A	10	8	80%
B	10	10	100%
C	10	10	100%
D	10	10	100%
E	10	7	70%
F	10	10	100%
G	10	10	100%
H	10	10	100%
I	10	10	100%
J	10	7	70%
K	10	10	100%
L	10	10	100%
M	10	7	70%
N	10	10	100%
O	10	10	100%
P	10	10	100%
Q	10	10	100%
R	10	10	100%
S	10	8	80%
T	10	10	100%
U	10	10	100%
V	10	10	100%
W	10	10	100%
X	10	10	100%
Y	10	10	100%
Z	10	8	80%
Total	260	245	94%

Figure 9. Performance of each Sign group of Alphabets.

amongst A, S and E, M. This happened because A, S and E, M look like each other in some samples.

For the words we choose some everyday words and tested then with our system and we got 92% accuracy see Figure 10.

English Word	Database Images	Recognised Images	Performance
HI	10	8	80%
BYE BYE	10	9	90%
DAD	10	9	90%
MOM	10	10	100%
GOOD NIGHT	10	10	100%
GOOD MORNING	10	9	90%
HOME	10	10	100%
SCHOOL	10	10	100%
THANK YOU	10	9	90%
HELP	10	8	80%
TOTAL	100	92	92%

Figure 10. Performance of each Sign group of Words.

The confusion while recognising the words is the way of producing the sign because not everyone does signs exactly the same way he/she did before. Each person has different hands in terms of hand size, finger size, shape and thickness. The major problem is to identify the universal pattern for a sign. We took samples from different persons, by using this approach we tried to overcome the problem and increase the efficiency of the system.

7 Conclusion

The goal of this work was to develop a practical real-life application to help remove barriers in communicating via ASL. A key aspect of our work was to develop a system that works effectively in real-time, and that requires minimum equipment such as a generic web-camera and a computer, laptop or similar. Previous work in this area is often restricted by the need for bulky hardware, probes, or cameras. The camera on a com-

puter, acts as the input source for our classifier. Using the Cross-correlation and edge detection approach we are able to use the input image stream to identify the start and end of ASL gestures. To facilitate this identification process, we require the user to produce a sign in front of the camera mounted on the computer.

7.1 Limitations

One limitation of the presented approach is its sensitiveness to the background scene. The background must be uniform because it can introduce additional structures during the edge-detection and binary image transformation stages that affect the classification if the background has many objects and glare, resulting in a miss classification of the sign. However, our experimental evaluation shows that the developed system is capable of achieving a classification accuracy of 92 to 94 % when identifying ASL gestures using real and synthetic data. Improvement of overall classification accuracy measurements while increasing the number of ASL gestures recognizable by our system can be extensions of this work. The following section summarizes the suggested research path to extend this work.

8 Future Work

We have presented a methodology to translate ASL signs to text and provided a prototype implementation of such system capable of translating basic ASL signs. Broadening the scope of this work, we present five alternatives of extending this thesis. Each of these are described in the following sections.

8.1 Learning Algorithms

By increasing size and resolution of the data, complexity of its management is also increased. As we discussed earlier the more training set for each letter or word we have the more accurate our results will be. A more flexible structure needs to be adopted, one that since no human movement is alike, must to be able to adapt. Such a management system is afforded by learning algorithms like neural networking.

8.2 Translation of Signs into Speech

Although the signs convey in most cases a literal meaning, they are not performed in the order in which English is articulated, thus a first requirement would be the translation of signs into English text. As mentioned, the expressions recognised would be instrumental in the shaping of signs into phrases. Once the signs have been translated into text, since speech synthesis has been developed quite completely, an off the shelf text to speech synthesiser would then be adopted to finally complete the translation.

8.3 Development of an English-to-ASL System

This system will include the entire processing architecture of the English to ASL translator. In which the dictionary words will be translated into signs. In this we will to use a speech recognition tool to convert voice to text and then those words will be represented by sign images.

8.4 Expression Identification

Recognition of the signs being conveyed gives the building blocks of the language, however the structuring of a phrase relies heavily on expression to shape it. To facilitate recognition of expression, the application of sensors to the face is not a practical option. A visual based technique would have to be adopted, using cameras with sensors and feature extraction techniques. The camera would need to be located on the portable system, giving rise to a non-oblique angle from which to apply the technique, thus requiring further processing of the acquired data such that an oblique perspective could be generated.

8.5 Video Calling with Sign translator

One of the biggest future plan is to implement a cloud based video calling interface where all the database storage and data processing will be done on cloud virtual machines. Also provide a desktop and mobile app (which you can install in your Android/IOS) so anyone can communicate with a deaf person without the help of a human interpreter.

The proposed system can also be integrated with current video chat environments like Skype or Google Hangouts. This can be achieved using the video calling API provided by Skype or Google Hangouts. Using the api we can get the video in and out stream and we can use them as our input data.

References

[1] https://www.microsoft.com/en-us/research/kinect-sign-language-translator-part-1/. *Microsoft*, 2006.

[2] http://www.washington.edu/news/2016/04/12/uw-undergraduate-team-wins-10000-lemelson-mit-student-prize-for-gloves-that-translate-sign-language. In , 2016.

[3] Y. Freund and R. Schapire. A decision-theoritic generalization of on-line learning and an application to boosting. In *journal of computer and system sciences*, Florham Park, New Jersey(USA), 1997.

[4] http://www.motionsavvy.com.

[5] Mathworks. Edge detection methods for finding object boundaries in images.

[6] Pramod K. Mishra Mokhar M. Hasan. Features Fitting using Multivariate Gaussian Distribution for Hand Gesture Recognition. In *Int. JComp Sci. Emerging Tech*, Uttar Pradesh (India), 2012.

[7] N. Otsu. A Threshold Selection Method from Gray Level Histograms. In *IEEE*, 1979.

[8] S.D.Lokhande V. S. Kulkarni. Appearance Based Recognition of American Sign Language Using Gesture Segmentation. In *(IJCSE) International Journal on Computer Science and Engineering* , Pune (India), 2010.

[9] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition*, Cambridge, MA(USA), 2001.