ISSN : 2350-0743

www.ijramr.com

International Journal of Recent Advances in Multidisciplinary Research

Vol. 03, Issue 07, pp.1649-1654, July, 2016

REVIEW ARTICLE

AUTOMATIC RECOGNITION OF BANGLA SIGN LANGUAGE USING ARTIFICIAL NEURAL NETWORKS (ANNS) FOR DEAF AND DUMB TO BRIDGE THE COMMUNICATION GAP

[1]Golam Kayas, *[2]Sakhawat Hossain [3]Himel and Mahraj Hasan

[1]Software Engineer Samsung R&D Institute Bangladesh Ltd. Dhaka, Bangladesh, India
[2]Department of Biomedical Physics and Technology, University of Dhaka, Bangladesh
[3]Department of Physics, University of Dhaka, Dhaka, Bangladesh, India

ARTICLE INFO

Article History:
Received 17th April, 2016
Received in revised form
30th May, 2016
Accepted 17th June, 2016
Published online 31st July, 2016

Keywords:

Computer Vision,
Open Natural interaction (NI)
Meachine Learning,
Artificial Neural Network (ANNs),
Back propagation,
Recognition of Bengali Sign Language.

ABSTRACT

Sign Language is the mode of communication among the deaf and dumb. However, integrating them into the main stream is very difficult as the majority of the society is unaware of their language. So, to bridge the communication gap between the hearing and speech impaired and the rest in Bangladesh, we conducted a research to recognize Bangla sign language using a computer-vision based approach. To achieve our goals we used Artificial Neural Networks to train individual signs. In the future, this research, besides helping as an interpreter, can also open doors to numerous conveniently other applications like sign language tutorials or dictionaries and also help the deaf and dumb to search the web or send mails more.

INTRODUCTION

Sign language is the mode of communication among the deaf and dumb. A common misconception about sign language is that it is universal, which is however not the case. In fact, sign languages, just like spoken languages are unique to a culture and have evolved over time. Moreover, they feature their own grammar and vocabulary and are generally acquired by deaf children as their mother tongue. Sign languages use manual gestures and body language to convey meaning. Static signs are generally used for alphabets and numbers where hand shapes define each sign. On the other hand, words and sentences are generally expressed through a combination of hand shape, orientation and movement of hands and arms. Additionally, facial expressions exhibit emotions and sometimes head movement, shoulder position, body posture and lip patterns are important parameters in expressing the meaning of a sign. Hundreds of sign languages are in use around the World today, some of which have not yet gained any legal acceptance. In Bangladesh, the Centre for Disability in Development (CDD) has developed a formal sign language for the Bengali deaf and dumb community, which is followed by schools for the speech and hearing impaired countrywide. Incorporating the deaf and dumb into the mainstream is difficult, mainly due to a lack of knowledge about sign language by the rest of the society. So, as to bridge this communication gap, scientists have been researching on methods to develop automatic sign language recognition systems. This field of research is still far behind and struggling. Moreover, research on recognition of Bangla sign language has not prevailed as it has for some other sign languages. So, our goal is to conduct a research to recognize Bangla sign language. There are several technologies that can be and has been employed in sign language or gesture recognition. For the purpose of our thesis we are using a computer-vision based approach with the help of Kinect Depth Camera and Neural Networks to recognize signs. For now, we are limiting recognition of isolated Bangla signs only and are considering only manual features for our experiment [1].

Corresponding author: Sakhawat Hossain
Department of Biomedical Physics & Technology, University Of Dhaka, Bangladesh, India

Sign as a Language

Components and Rules

Hands are the basic means of communicating using sign languages. Hand shapes, hand movement, palm orientation, and hand position are some of the most important components to convey the meaning of a sign [8]. However, signs are not confined to manual features, i.e. hands and arms only. Non-manual features, such as head position, head tilt, body posture, eye movement and lip shapes are also important parameters used in conveying meaning of a sign [2]. In fact, facial features are very important to express emotions. Some of these features

are also important to differentiate between questions, negations and affirmations. Some signs are shown using both hands, while others with only one hand. The right hand is generally called the dominant hand and is used to convey almost all signs unless the signer is left-handed. Signs can be either static or dynamic. In dynamic signs using two hands, both hands can move or one might be static while the other is in motion. In such a case, it is generally the dominant hand that is in motion, while the other is at rest. If both hands are moving simultaneously in a sign, it is important that the hand shapes both hands are same, but if only one hand is moving at a time in a two-hand sign then the shapes of the two hands can differ [8][2]. One important fact to consider about sign languages is that, same hand shapes or same hand motion can be used to express different signs. For example, in Bangla sign language, window and clean have the same hand movement but the hand shape for the two signs are different. Similarly, picture and table are signed using the same hand shape and palm orientation, but different hand movement. The images of Bangla signs of window and clean and picture and table, illustrating their similarities and differences, are given in Fig. 1 and Fig. 2 respectively.

Fig. 1. Bangla Sign for Window (left) and clean (right) (Figure taken from [8]) Same hand motion (same direction). Different hand shape

Fig. 2. Bangla Sign for Table (left) and Picture (right) (Figure taken from [9]) Same hand shape. Different hand motion (different plain)

Another very important feature to consider is the signing space. All signs are generally produced within this space. The signing space includes the area around the head, the belly and both arms.

Bangla Sign Language

In Bangladesh a formal sign language has been established only recently. In the year 2000, Center for Disability in Development (CDD) took the initiative to standardize communication with sign languages in this country. Before this step, there were different local variants and no national dialect existed. CDD has published many books rich in vocabulary and grammatical rules and they also provide sign language training in their training center [9]. Other than CDD, there is only one high school for deaf children in Bangladesh, Dhaka Bodhir High School. People in Bangladesh still are ignorant of this mode of communication and thus the deaf children still cannot lead an uncomplicated life here. However, measures are being taken in order to aware the people of Bangla Sign Language and we can hope that eventually life will become much easier for the hearing and speech impaired in the future.

Kinect

Kinect was developed by Microsoft and Prime Sense and was released on November 2010. Kinect combines an RGB camera, a depth sensor and a multi array microphone. The best feature of the Kinect camera is its depth sensor, which uses an infrared projector and a CMOS sensor and is capable of tracking users in 3D independent of the lighting condition [3]. Initially developed for the Xbox 360 video game console and windows PC, the Kinect camera is now being used by the computer vision community and many programmers as they realized that this depth sensing technology could be used for many purposes other than gaming [2][3]. Currently, there are three software frameworks available for Kinect, Microsoft SDK, Open NI and Open Kinect developed by Microsoft, Prime Sense and the hacker community respectively. In 2011, Microsoft and Prime Sense released their versions of frameworks for Kinect. But, before them, the hacker society came up with Open Kinect by reverse engineering the USB stream of data from the Kinect device. These frameworks led to the possibility to develop non-commercial products and thus presenting the computer science community with an excellent and cheap tool to build and research on different computer vision technology [3][2].

Open NI and Processing

The Open NI or Open Natural Interaction driver, released by Prime Sense, has a feature-rich open source framework and can be combined with closed source middleware called NITE for user skeleton tracking and hand gesture recognition [12]. In our research we have used the Simple Open NI library and it is a wrapper for Processing. Processing is an open source programming language and environment, which helps create images animations and interactions. The language builds on the Java language, but has a simplified syntax and graphics programming model [13]. Processing also provides a Java wrapper and has been used for our project.

Artificial Neural Networks (ANNs)

Artificial Neural Networks (ANNs) are non-linear mapping structures based on the function of the human brain. They are powerful tools for modeling, especially when the underlying data relationship in unknown. ANNs are being used to solve problems, such as logistic regression, Bayes analysis, multiple regressions etc. Input nodes take input of the system; according to which the output of the ANN is generated. Usually there is only one input layer. There is also only one output layer in ANN usually. Output nodes are equal to the number of the outputs of the system. There is no magic formula to select optimum number of hidden nodes. However some thumb rules are available to calculate the number of hidden nodes. Three later fundamental networks widely used today are:

A. Back Propagation network

This is a multi-layer Perceptron-based ANN, whichgives an elegant solution to hidden-layers learning (Rumelhart, 1986 and others).

B. The Hopfield Network

This network was introduced by John Hopfield in 1982. It is different from the earlier four ANNs in many important aspects, especially in its recurrent feature of feedback between neurons and hence, it is to a great extent a separate ANN-class in itself.

C. The Counter-Propagation Network

Proposed by Hecht-Nielsen in 1987, it utilizes Kohonen's Self-Organizing Mapping (SOM) to facilitate unsupervised learning.

Back Propagation Learning Procedure

Back Propagation network is often considered to be the classic ANN. However, it is less of network and more training or learning algorithm. The network used is generally of the simple type and are called Feed-Forward Networks or occasionally Multi-Layer Perceptrons (MLPs). A Back Propagation network learns by example, i.e. example of what is required from the network is provided to the algorithm, which changes the network's weights so that when training is finished, it will produce the required output, which is known as the Target, for a particular input. Once the network is trained, it will provide the desired output for any of the input patterns. Back Propagation networks are ideal for simple Pattern Recognition and Mapping Tasks. Fig. 3 shows how a Back Propagation Network works.

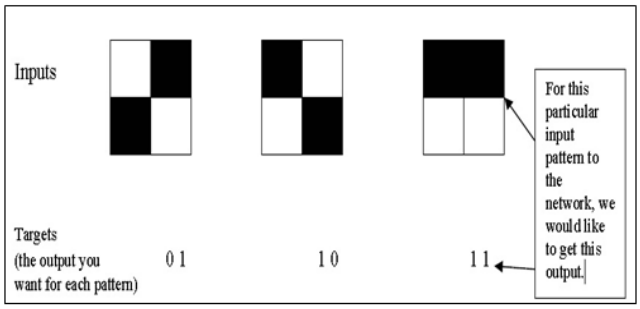

Fig. 3. Back Propagation Network

In a Back Propagation Network, the network is first initialized by setting up all its weights to be small random numbers, for example between –1 and +1. Then the forward pass is applied, i.e. the input pattern is provided and the output is calculated. As all the weights are random, the output provided initially is completely different from the Target. The Error of each neuron is then calculated, which is essentially the Actual Output subtracted from the Target. We have chosen the vision- based approach for recognizing signs as opposed to data gloves or other exotic techniques. The vision-based approach is a more natural process and is less intrusive to the signer. Our initial goal is to recognize isolated signs conveyed through movement of hands. Thus, for the purpose of our thesis we will ignore facial expressions and other parameters involved in signing. The entire recognition process can be broken down into three main phases:

- Tracking
- Feature Extraction
- Training

Before starting our experiment we investigated several techniques and tools. At first we considered hand tracking using a web-cam. However, after extensive training using Open CV's haartraining to track hands, the accuracy was still very low.

We figured that due to the variety of skin complexion of Bangladeshi people and the possibility of various hand orientation and shape, this technique would not be efficient.Then we came across the OpenNI framework, which provides high quality tracking methods and thus we used the SimpleOpenNI library with Java for the first two phases. For the third phase we chose to use Artificial Neural Networks implemented in MATLAB.A step-by-step process we followed to conduct the experiment in order to validate our research is given in the form of a flowchart in Fig. 4

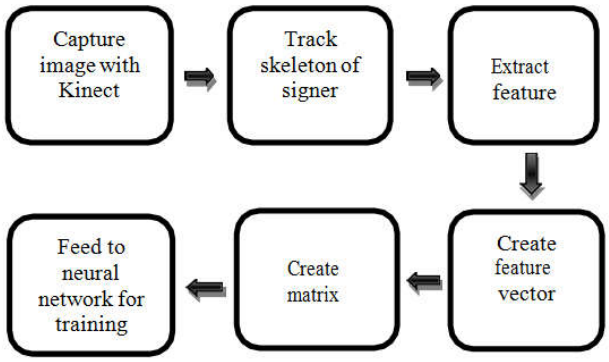

Fig. 4. Step-By-Step Recognition Process

Tracking

We tracked the signer skeleton, which provides joint information of the entire body. As we focused on manual features only, tracking the hands was the most important task. Besides the hand we also needed to track the head, neck, torso and elbows.However, it requires controlled environment and good lighting conditions. Meanwhile, Kinect is capable of tracking a user's full body or hands independent of lighting conditions or other environmental variables. The SimpleOpenNI library provides built-in functions to track the skeleton and also to extract coordinates of the joints. The result of Kinect and OpenNI tracking a human skeleton using the depth sensor is shown in Fig. 5

Feature Extraction

Each sign is associated with a set of features that need to be extracted in order to distinguish one sign from another.Since we are only focusing on manual gestures; we will extract features related to hands only. Our initial aim was to extract the hand shape along with other features. However, as stated before, individual finger recognition is a limitation of the frameworks available for Kinect.

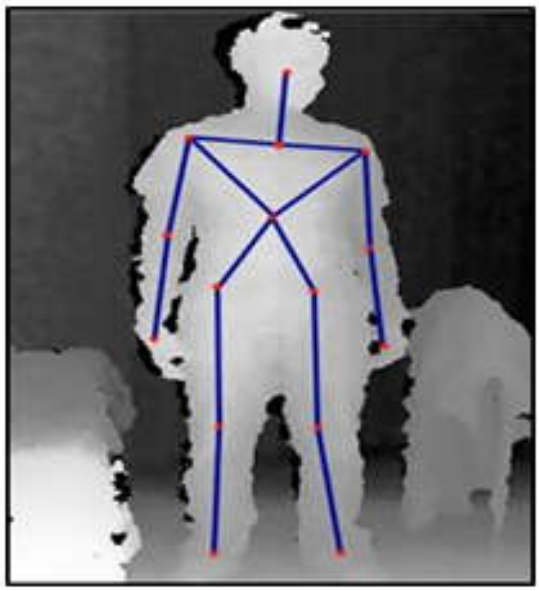

Fig. 5. Skeleton Tracking

Though, some hand shape information could have been obtained by customizing the existing API, it would have taken more time than we have to complete our research. So, we decided to proceed without extracting hand shapes for now. We extracted seven features, which were also extracted in the recognition of German Sign Language in [1], however they adopted a different training method than ours and they also considered the neck as the reference point for all calculations, while we took the head as the reference point. The seven features and the calculations are given below.

Two dimensional position of each hand relative to head (xAbs, yAbs)

$xAbs$=| $andX$ $eadX$|
$yAbs$=| $andX$ $eadX$|

Where, ($andX$, $andY$)and($eadX$, $eadY$) are the 2D coordinates of the hands and head respectively.

Distance between both hands (distance)

$Distance$
=√(($rig\ t\ andX\ left\ andX$)^2+($rig\ t\ andY\ left\ andY$)^2)

Two dimensional movement of each hand or Position of each hand relative to position of hands prior to the last two updates (xrel,)

$xrel$=| $andXprev$ $andX$|
$yrel$=| $andYprev$ $andY$|

Where, ($andXprev$, $andYprev$) are the 2D coordinates of hands two updates before.

Absolute velocity of each hand(v)

v=√$xrel^2$+$yrel^2$

Absolute distance of each hand from head(d)

d=√$xAbs^2$+$yAbs^2$

Two dimensional normalized velocity of each hand(vx.vy)

$$(vx, vy = \left\{ \begin{matrix} (0,0), xrel = 0\ and\ yrel = 0 \\ \left(\frac{xrel}{xrel + yrel}, \frac{yrel}{xrel + yrel}\right), ot\ \ erwise \end{matrix} \right\}$$

Positionof bothelbowsrelative toneck (ex,ey)

(ex,ey)=($|elbX–neckX|$,$|elbY–neckY|$)

The region above the torso was considered as the signing space and the updates were recorded and features were calculated in each frame only if the dominant hand, i.e. the right hand was above the torso, i.e. y-coordinate of right hand was greater than that of the torso. All these features were combined into a feature vector(f), which was then used in the training phase. The feature vector is given below:

f=($xAbs$,$yAbs$,$xrel$,$yrel$,vx,vy,ex,ey,v,d,$distance$)

A frame containing all extracted features of both the hands combined together in a particular frame is shown in Fig. 6. It is basically a vector with 21 values.

[253.04129 610.7383 201.94067 581.78613 10.866272 98.85979 29.40271 101.32846 0.09903091 0.9009691
0.22490971 0.7750903 204.25998 225.66953 211.0105 200.09103 99.455185 105.50818 661.0833 615.8369 455.9022]

Fig. 6. A Feature Vector for a Single Frame

Training

Basically, each frame has 21 values to describe the features we have extracted. A set of vectors will represent a particular sign. If the signer gives the sign quickly then there will be less number of frames in a sign and thus less vectors, whereas if the sign is given slowly number of vectors representing the sign will increase. To avoid this problem we selected 30 random frames including the first and last frame from the vector of the frames to describe a sign. As a result, a sign is described by a 21 X 30 matrix, i.e. 30 frames each containing 21 values. Then we transformed this matrix into a 7 X 10 matrix. To make this transformation we divided the 21 X 30 matrix into seventy 3-by-3 grids. Each grid was replaced by one value, which is the average of every value inside that grid. An example of a conversion of a 6 X 6 matrix into a 2 X 2 matrix using the same technique is given below.

6×6 Matrix

2	3	2	4	5	6
1	2	2	3	7	5
3	2	1	6	4	5
1	2	3	6	4	8
1	3	2	6	7	8
2	1	3	8	6	7

2×2 Matrix

2	5
2	6

4×1Matrix

2
2
5
6

After getting a 7×10 matrix for each sign we make it a 70×1 matrix. For Example the 2×2 matrix above will become 4×1 matrix as shown above. We created this system for five Bangla signs (Brother, Tea, Chair, Door, Notebook) used in daily lives and taught to primary deaf school students. For each sign we had 10 samples and two inexpert signers who learnt these signs specifically for the purpose of this thesis conducted the signs for training. So finally we had ten 70 X 1 matrix and

we created one 70 X 10 matrix from these 10 matrices. We created a Back Propagation Artificial Neural Network (ANN) with 3 layers (one input layer with 70 nodes, one hidden layer with 48 nodes, one output layer with 5 nodes). Then each column of the 70 X 10 matrix created earlier was fed as input to our ANN for training the system.

Experimental Results and Evaluation

We calculated our accuracy using the following variables,

$$\text{Precision} = \frac{|\{releavent\ documents\} \cap \{retrieved\ documents\}|}{|\{retrieved\ documents\}|}$$

$$\text{Recall} = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{relevant\ documents\}|}$$

Recall and Precision can be defined using the following variables

- ***True Positive (tp):*** True positive means number of correct results we are looking for.
- ***True Negative (tn):*** Means correct absence of the irrelevant result in our system's result.
- ***False Positive (fp):*** Means wrong result in our system's output (Unexpected output).
- ***False Negative (fn):*** Means missing expected outputs, i.e. the result should be included in system output but not there.

RESULTS OBTAINED

We divided the input of our system in three parts:

- Training data, which we used to train the system given as Input (8 samples for each sign).
- Testing data created by the same signers who signed the test data for training (2 samples for each sign)
- Testing data generated by new signer (10 samples for each sign)

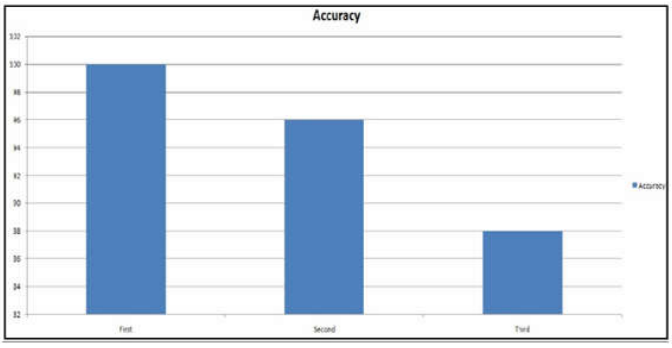

Fig. 7. Accuracy comparison of three types of input

The values of tp, tn, fp and fn we obtained for each type of input are given below and the corresponding values of precision, recall, true false rate and accuracy are also calculated.

First case

TP	TN	FP	FN
10	40	0	0

Precision = 10/(10+0) * 100= 100% Recall = 10/(10+0)*100 = 100%
True False rate = 40/(40+0)*100 = 100%
Accuracy = (10+40)/(10+40+0+0)*100 = 100%

Second case

TP	TN	FP	FN
9	39	1	1

Precision = 9/(9+1) *100 = 90%
Recall = 9/(9+1) * 100 = 90%
True False rate = 39/(39+1)*100 = 97.5%
Accuracy = (9+39)/(9+39+1+1)*100 = 96%

Third case

TP	TN	FP	FN
7	37	3	3

Precision = 7/(7+3) *100 = 70%
Recall = 7/(7+3) * 100 = 70%
True False rate = 37/(37+3)*100 = 92.5%
Accuracy = (7+37)/(7+37+3+3)*100 = 88%

The accuracy we obtained during our experiment is given below in terms of separate type of inputs.

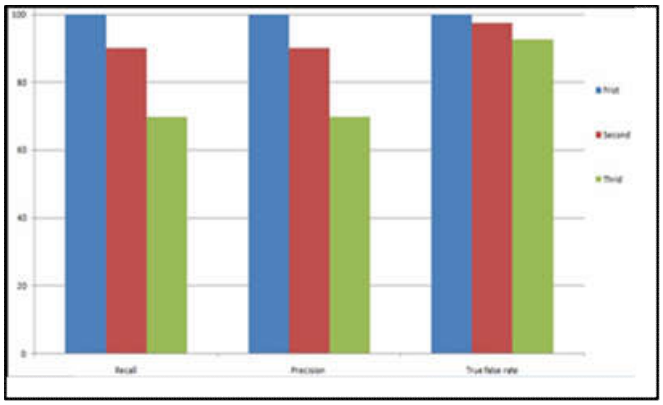

Fig. 8. Recall, Precision and True False Rate comparison of three types of input

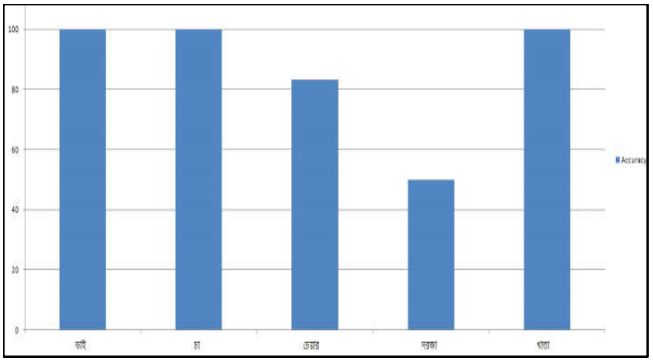

Fig. 9. Accuracy for individual signs

Evaluation of Results

We can observe in Fig. 7 that when given the training data as input the recognition rate is 100 percent, meaning that the training was successful. Then when a separate test data was given to the neural network, the accuracy rate was 96 percent.

The same signers who signed for the training sample provided this data. When the third input is given to the network the accuracy rate decreases to 88 percent. Thus, it cane be seen that the system performance becomes worse when the input data is provided by signers the system does not know. Also from Fig. 8 we can see that the recall, precision and true false rate is 100 percent and above 90 percent for the first and second type of input respectively. For the third input type recall and precision is just about 70 percent. In Fig. 9 we can see the accuracy rate of individual signs. The accuracy rate of brother (“ ”, 1st from left), tea (“ ”, 2nd from left) and notebook (“ ”, last from left) are 100 percent each. However, chair (“ ”, 3rd from left) and door (“ ”, 4th from left) have very medium (85 percent) and low (50 percent) accuracy rate respectively.

Fig. 10. Notebook and chair (figure taken from [9])

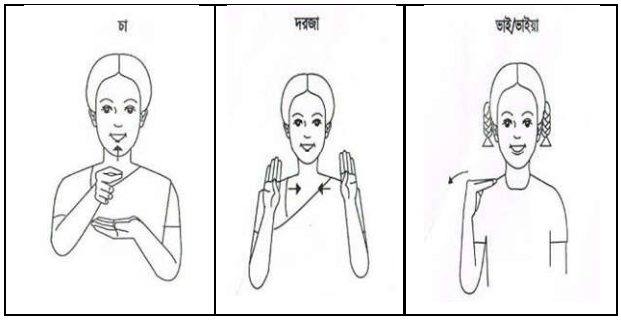

Fig. 11. Tea, Door and Brother (figure taken from [9])

Conclusion

Our experiment spans over a very small subset of automatic sign language recognition. There is a lot more that needs to be and can be done in this field to achieve the final goal of reducing the communication gap between the hearing impaired and the rest. We need to improve the work we have done in our thesis to make recognition of isolated signs more accurate. Firstly, features related to hand shapes have to be extracted, otherwise signs with similar hand movements and position will be difficult to separate using the existing features that has been extracted. The next step will be to increase the number of training samples and to try training methods other than neural network to analyze which one produces the best result for Bangla Sign Language recognition. Also, the numbers of trainers need to be increased and this will improve the rate of recognition. After improving on the current work we have done so far, our next aim will be to research on extensions to the project. At first, recognition of both static and dynamic signs need to be incorporated within the same system, which has not been achieved to greater accuracy yet, especially for Bangla Sign Language. Then continuous sentences need to be recognized and mapped to the corresponding spoken grammar. Also, text-to-sign and sign-to-text systems need to be combined together in one system to make the communication a two way process. In the future, continued research in this area, besides helping as an interpreter, can also open doors to numerous other applications like sign language tutorials or dictionaries and also help the deaf and dumb to search the web or send mails more conveniently.

Acknowledgment

As mentioned before we have used five signs to measure the accuracy of our method during the research. Information on how to perform these signs is given below. All signs shown here belong to the Bangla Sign Language vocabulary only.

REFERENCES

“Scientific Understanding and Vision Based Technological Development for Continuous Sign Language Recognition and Translation.” SignSpeak Project, Annual Public Report. http://www.signspeak.eu/

Andersen, M.R., T. Jensen, P. Lisouski, A.K. Mortensen, M.K. Hansen, T. Gregersen and P. Ahrendt. Kinect Depth Sensor Evaluation For Computer Vision Applications. Department of Engineering , Aarhus University, Aarhus University, 2012.

Dewan Shahriar Hossain Pavel, Tanvir Mustafiz , Asif Iqbal Sarkar, M. Rokonuzzaman. "Modelling of Bengali sign language expression as dynamic 3D polygons for developing a vision based intelligent system for dumb people”National Conference on Computer Processing of Bangla. 2004.

Golam Kayas and NajeefaNikhat Choudhury,”Automatic Recognition of Bangla Sign Language” BS thesis,School of Engineering and Computer Science,Department of Computer Science and Engineering, BRAC University. http://dspace.bracu.ac.bd/bitstream/handle/10361/2387/Automatic%20Recognition%20of%20Bangla%20Sign%20Language. pdf.pdf?sequence=1

Kaushik Deb, Helena Parvin Mony & Sujan Chowdhury. "Two-Handed Sign Language Recognition for Bangla Character Using Normalized Cross Correlation." (Global Journals Inc. (USA)) 12, no. 3 (February 2012).

Kinect Fact Sheet, Microsoft News Center. June 2010. http://www.microsoft.com/presspass/presskits/xbox/docs/KinectFS.docx.

Lang, Simon. Sign Language Recognition With Kinect. Institut for Informatik, Freie Universitat Berlin, Berlin: Freie Universitat Berlin, 2011.

Nahid Sultana Juthy, Broj Gopal Shaha, Md. Sharafat Ali Shojol. Ishara Bhashay Jogajog (Communicating through Sign Language). Dhaka: Center for Disability in Development, 2005.

OpenNI. http://openni.org/

Parton, Becky Sue. "Sign Language Recognition and Translation: A Multidisciplined Approach From the Field of Artificial Intelligence." *Journal of Deaf Studies and Deaf Education (Oxford Journals)* 11, no. 1 (2005).

Precision and recall: http://en.wikipedia.org/wiki/Precision_and_recall

Processing. http://processing.org/

Sarella, Kanthi. Formulation of an Image Processing Technique for Improving Sign2 Performance. Final Report.

Sign Language. http://en.wikipedia.org/wiki/Sign_language.

****