Intelligent Sign Language Verification System – Using Image Processing, Clustering and Neural Network Concepts

Foez M. Rahim[1], Tamnun E Mursalin[2], Nasrin Sultana[2]

[1] Student, American International University-Bangladesh, [2] Faculty, University Of Liberal Arts-Bangladesh
email: foez@hotmail.com, temursalin@gmail.com, nsultana_du@yahoo.com

Abstract: The paper uses image processing system to identify, especially Bengali alphabetic sign language used by the deaf people to communicate. Initially the data is processed from raw images and clustered to accommodate similar pattern images into a clustered database. Later, two type's data were fed into the neural network: the processed image object and the clustered data to train the network. The system successfully identified the sign languages 99% of the time. The system is tested on Bengali sign language but it can detect any sign language with prior image processing and clustering.

Keywords: clustering, neural network, image processing, sign language

1. INTRODUCTION

Dumb people are usually deprived of normal communication with other people in the society. It has been observed that they find it really difficult at times to interact with normal people with their gestures, as only a very few of those are recognized by most people. Since people with hearing impairment or deaf people can not talk like normal people so they have to depend on some sort of visual communication in most of the time.

Sign Language is the primary means of communication in the deaf and dumb community. As like any other language it has also got grammar and vocabulary but uses visual modality for exchanging information [1]. The problem arises when dumb or deaf people try to express themselves to other people with the help of these sign language grammars. This is because normal people are usually unaware of these grammars. As a result it has been seen that communication of a dumb person are only limited within his/her family or the deaf community.

The importance of sign language is emphasized by the growing public approval and funds for international project. At this age of Technology the demand for a computer based system is highly demanding for the dumb community.
However, researchers have been attacking the problem for quite some time now and the results are showing some promise. Interesting technologies are being developed for speech recognition but no real commercial product for sign recognition is actually there in the current market [2].

So, to take this field of research to another higher level this project was studied and carried out. The basic objective of this research was to develop a computer based intelligent system that will enable dumb people significantly to communicate with all other people using their natural hand gestures. The idea consisted of designing and building up an intelligent system using image processing, data mining and artificial intelligence concepts to take visual inputs of sign language's hand gestures and generate easily recognizable form of outputs.

1.1 OBJECTIVE AND SCOPE

As it was mentioned above, the primary objective of this research project was to develop an intelligent system which can act as a translator between the sign language and the spoken language dynamically and can make the communication between people with hearing impairment and normal people both effective and efficient. Beside this, the designing of the system was focused on the dynamic association of its language domain, i.e. it can deal with any sign languages. The vision was to research and develop an engine or methodology for sign language recognition rather than just only output generation; so that further research can be done on this model / procedure and a fully operational and commercial system can be built for gesture recognition, to fulfill the communication requirement of the deaf community.

Although the system is designed to be able to deal with any sign language, but to achieve the project's goal and carry on with the research Bengali sign language was selected as the language domain for the moment, which is a standard procedure for dumb people to communicate.

The chart below shows the sign representations of Bengali vowels (Figure 1) and some of the consonants (Figure 2) used from the sign language domain to justify the system's performance.

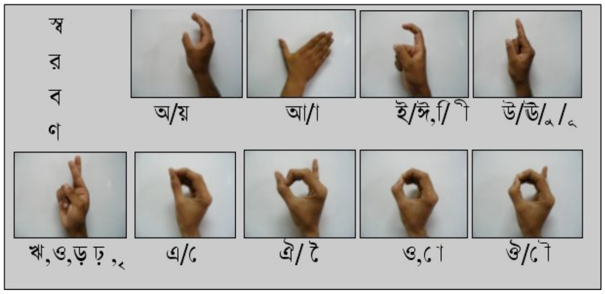

Figure 1 : Bengali vowels in Sign Language

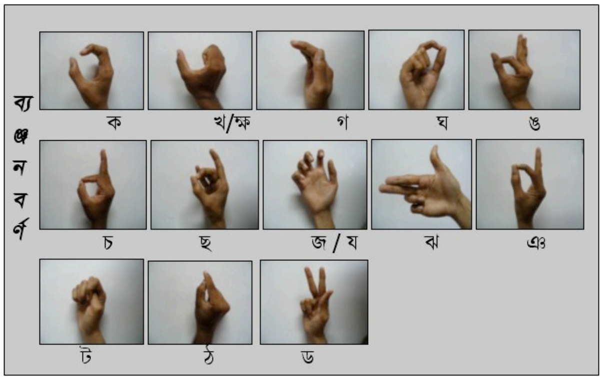

Figure 2 : Some Bengali consonants in Sign Language

Different instances of each of these images are used as inputs to carry out the project described through out the paper.

2. LITERATURE REVIEW

Not many Researches have been carried out in this particular field, specially in Bengali Sign Language Recognition. Few researches have been done on this issue though and some of them are still operational, but no body was able to provide a full fledged solution to the problem. Some of these earlier approaches are mentioned through out this section.

In one of the previous approaches, a research has been carried out and a system was developed by M. Rokonuzzaman [3] which analyzes video clips of different gestures of sign languages taken as input and gives a regular language expression as an audio output. In the technological approached used, angles of different parts of the hand with body were calculated manually by analyzing captured images from input video frames. These angles along with corresponding audio meanings were also previously stored manually in a database or file. The system then matches the calculated angles with the database and the corresponding audio file is played in the speaker of the pocket PC.

The limitations that both of this research possessed is firstly, the system recognizes and displays only single handed gestures. Secondly, actual frame rate of the animation is too quick for interpreting the sign language for which the frame rate was decreased manually. Next the angles had to be calculated manually and thus the database has to be created for every set of signs.

Another system named as "*Intelligent Assistant*" human machine interface [4] was developed in the late 2002 to communicate the deaf people. This system could understand ten Bengali expressions and was able to instantly transform the expression into sign language. The objective of the approach that was chosen to build up this system was to understand the Bengali expression and produce related sign language for deaf or hard hearing people. The system used Microsoft's Voice Command and Control Engine (which actually did most of the part) for capturing sound input with the help of a microphone and converting it into text. The system then matches the text with the list of word previously stored in its knowledge base. Once the word matches the database, it displays the graphical output of that word with 3D graphical hand images, which were simply mapped also according to the text. With only ten words it produced an output accuracy of no more than 82%.

The main drawbacks of this system were – it could not perform efficiently in noisy environments as it also included noise with the input resulting a wrong output generation, it showed signs using only one hand where as there are some signs that require both hands, it could not recognize a complete sentence and finally, different pronunciations of a word resulted in wrong outputs.

An Interesting attempt was taken in using Image Processing and to some extent Neural Networking for the recognition of Bengali Sign Language by the AIUB students in their thesis project [5]. But they failed to generate the output using the Neural networking, due to memory overflow. Instead the output of the system they developed was generated using 3D graphics tools.

These sorts of system were also built for other domains where a computer take voices as input, do digital sampling and maps an acoustic speech signal to some form of abstract meaning. [6]

The intelligent system that is described in [7] is a more technical and complicated system compared to the above mentioned systems. In this case the signs were shown single handedly wearing a glove containing different dots in each finger. The signs were then taken as inputs as digital photos captured in real time. The program then analyzed the dots of the graphics in the image file to understand what sign has been shown and then output the interpretation of the sign in regular language in both written and audible form which were pre-recorded as wave files. The technique involved clustering of the dots and mapping the results of this clustering to pre-defined tables.

This system was only limited to Bengali 1-10 numbers which actually does not require any intelligent system to understand. Moreover the dots were only marked at the front face of the fingers and the fingers had be spread as far as possible from each other to work correctly, but there are lots of signs representing Bengali alphabets in which fingers are closely compact and they are not front faced in most of the time, even in some cases they overlaps one another. So the clustering technique used in this system is fairly capable of dealing with those cases. Noisy dots in the environment or the pictures could also affect the resulting output of this system.

Later, State Machine based approaches were also used to resolve the sign language recognition problem. The research project carried out in [8] was an extension of the previous one [7], where state machine was implemented. The gesture capturing module of the system used the same input setup as before with camera and gloves. With the video inputs, the system was designed to determine the mark positions of the fingers in each frame and pass it to the state machine. For recognizing gestures the state machine then process the data and decides the final gesture made once it reaches to the final state. The output is displayed in textual, audible and visible form.

The main drawback of the system was its complexity. The state machine was designed to interpret only four gestures. It needs to be re-designed to include other gestures. It lowers flexibility of the system. Other limitations are same as the previous system.

The use of Neural Networking was another approach used to recognize signs. Sandberg [9] provided a system using a combination of Radial Basis Function Network and Bayesian Classifier to classify a hybrid vocabulary of state and dynamic hand gestures. A Neural Networking used system exists [10] which classifies Japanese Sign Language. The work in [11] presented an application for learning sign language where some intelligence were observed with the help of Neural Networking procedures.

Perhaps the most interesting work in this area was Thad Starner's Master's Thesis in recognizing American Sign Language using Hidden Markov Model [12]. He used a camera looking at a singer whore wore two colored gloves. Approximately 40 signs were chosen and data over 500 sentences were captured, each sentence containing an average of 4 signs. The camera was then connected to a machine with special hardware for processing video that would extract the position information about each hand. Though the system was very expensive but worked quite well and produced a raw classification rate of about 95%.

Unlike any other approaches in recognizing the Bengali sign languages used by deaf and dumb people, the intelligent system that is developed and described all though out this paper used Image Processing for pattern recognition, Data mining Clustering techniques and Artificial intelligence Neural Networking techniques combined to accomplish the job successfully. Almost all the systems described above used some kind of tools to achieve the goal of sign recognition, so they were platform dependant. On the other hand as the proposed system was fully developed in Java code, it has the advantage of platform independency. Not only these, but it is also capable of dealing with two hands in exception with most of the above systems or even it is able to deal with other body parts according to the training. More over this intelligent sign recognition system create its own database automatically in its training session which makes it dynamic and can also be used for other sign languages equally rather than Bengali Sign Language only.

3. METHODOLOGY

The proposed system works in two phases to perform the sign language verification. First in the Training session it trains the system with sample of raw data from the domain to form clusters. Secondly, once the system is trained it can be used to classify random images of the trained language according to the clusters. To accomplish this job, image processing, clustering and Neural Networking of data mining and Artificial Intelligent techniques were used.
The block diagram in figure 3 gives an overview of the whole system.

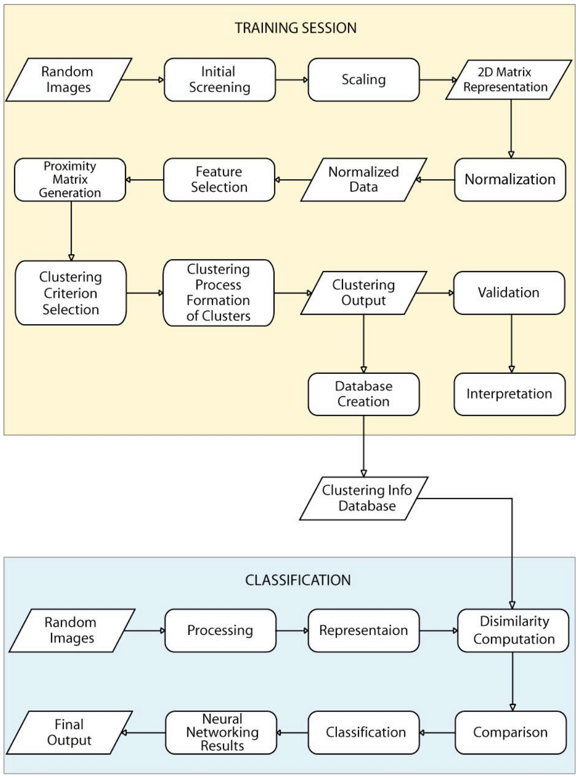

Figure 3: Block Diagram of the Proposed System

The system is called an "Intelligent System" due to fact that it can train itself automatically and creates a knowledge base for a given domain of sign language. Once trained the training results can be used further to identify hand gestures as long as the domain remains unchanged. For this reason a training session is required, where first a sign language is chosen as the domain, then a set of data from that domain is fed to the system. These data are the sample hand gesture images which are processed, screened and represented in suitable form for the clustering process. In the clustering process separate clusters are formed from the sample raw data, which actually represents different signs with corresponding meanings. The result of the clustering process is shown in both textually and graphically for verification. Another output of the whole training session is a database where information of each clusters or classes made is stored.

As mentioned earlier the training process is carried out only once for a particular domain while the next phase, that is, the Classification process is a continuous process. In this phase, the system takes random images and the database containing clustering information as inputs. Image processing techniques are used to process the input images as before to generate useful data for the Neural Networking process. These data are finally classified by the Neural Networking process according to the clusters, producing the desired output. The output is shown both in visual and textual form in which every signs are verified and displayed along with its corresponding meanings.

4. IMPLEMENTAION

The proposed methodology described in section 3 was implemented in this research project and an efficiently working "Intelligent System" for sign language verification was developed. At this particular project the selected domain was ten different signs from the Bengali Sign Language. A sample total of 24 different image instances of these signs were taken as the working input of this system (figure 4).

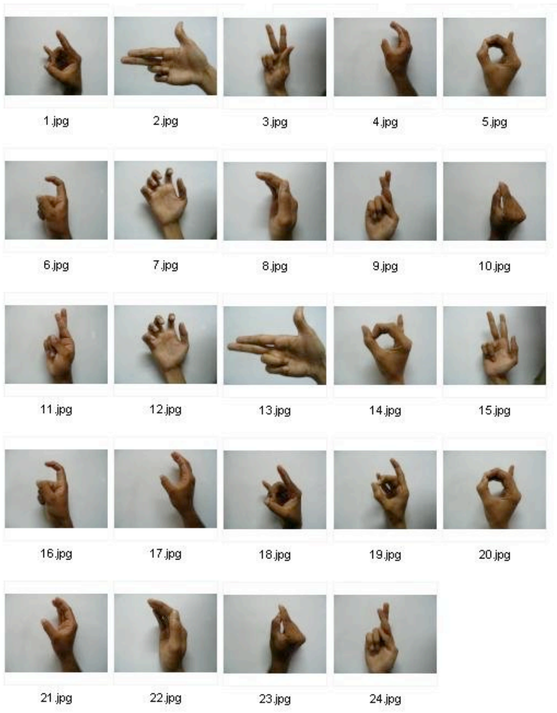

Figure 4: Experimented Input Set

All the separate processes of this system regarding these inputs are explained below.

4.1 IMAGE PROCESSING

As the inputs are raw images, for both the training and classification session, they need to be converted into suitable form of data before any sort of calculation can be done on them. So the first step involves some image processing. Images, when fed to the system are first rendered, then scaled to 800*600 pixels for standardization and finally converted into corresponding 800*600 2d matrixes according to the pixel colors (figure 5).

800 * 600 matrix

6315100.0	6249307.0	……………….	6051928.0
4933963.0	5125810.0	……………….	6249307.0
. . .	. . .	……………….	. . .
6174725.0	3889534.0	……………….	5048017.0

Figure 5: Data Matrix representation of each image

Each of these matrixes is then normalized to a single one dimensional array of 480000 in length. The above sub-processes are followed for every sample images. The final output of this phase is again a 2d matrix, but this time it is composed of all the one dimensional arrays, where each row represents a picture with its corresponding data. In this particular experiment the size of this output matrix of the image processing phase was 24*480001 as shown in the figure below, where the first column represents the image number.

24 * 480000 matrix

1	6315100.0	6249307.0	…………	5048017.0
2	5112958.0	4984327.0	…………	6433801.0
. . .	. . .	. . .	…………	. . .
24	7758947.0	7448201.0	…………	8011519.0

Figure 6: Normalized Data Matrix representation of all the input images

This representation was very much important as the input for the upcoming clustering or neural networking processes.

4.2 CLUSTERING PROCESS

The basic task of the training session is to take random images from the domain and make groups or clusters among them, so that each cluster represents a single meaning. This means that the images underlying the same clusters are similar and they represents the same alphabet or expression. The average values of all the images in a cluster are examined, calculated and stored inside a database. These values represent the center of each cluster and the final database contains the set of all clusters loading the system with a domain.

The main part of the training session is achieved by the Clustering process. Clustering is a very useful and efficient data mining widely used technique that was developed in the necessity of separating a bunch of data into groups whose members belong together. Each object is assigned to the group it is most similar to. Clustering does not require a prior knowledge of the groups that are formed and the members who must belong to it. Clustering techniques discover groups of similar instances, instead of requiring a predefined classification. So it can be considered as the most important unsupervised learning problem. In this case of clustering, the problem is to group a given collection of unlabeled patterns into meaningful clusters. Thus clustering is a task that identifies a finite set of clusters to describe the data. Clustering algorithms divide a data set into natural groups (clusters). Instances in the same cluster are similar to each other; they share certain properties and are dissimilar to the objects belonging to other clusters [13-17].

After the image processing is completed for the input images, the normalized data are fed to the clustering process. K-mean clustering algorithm was used as the clustering strategy. It is selected due to its efficiency in hard clustering [17-20]. Other clustering algorithms like fuzzy clustering or hierarchical clustering does not fit this particular problem. Fuzzy clustering creates overlapping clusters [21, 22] where as Hierarchical clustering generates a tree like structure called 'Dendogram' as the output [23]. None of these are suitable for this project, where hard partitioned clusters with concrete data are required for analysis. This is why K-mean clustering was used for this project.

The basic steps of K-mean clustering algorithm are shown as follows [24]:

1. Specify k, the number of clusters to be generated.
2. Choose k points at random as cluster centers.
3. Assign each instance to its closest cluster center using Manhattan distance.
4. Calculate the centeroids (mean) for each cluster; use it as new cluster center.
5. Reassign all instances to the closest cluster center.
6. Iterate until the cluster centers don't change any more.

Like the algorithm suggests the clustering process works in several steps. It takes number of clusters, k, to be formed and the number of iteration to be performed as input along with the image data matrix. Data are first extracted according to some feature selection and k clusters are formed. At the first iteration, each cluster was assigned with a single image data randomly, which were considered as the center of that cluster. From this a center matrix is generated for all the k clusters.

1	6315100.0	6249307.0	…………	5048017.0
2	5112958.0	4984327.0	…………	6433801.0
. . .	. . .	. . .	…………	. . .
k	3133579.0	3095404.0	…………	4381004.0

Figure 7: Center Matrix

Proximity measures are taken after that. In this case 'Manhattan Distance' was used as the dissimilarity measure. So, the Manhattan distance for every pixel of other images with respect to the corresponding points of all the clusters were found and placed in a 'distance matrix' (figure 8). The first row represents the image number and the first column represents the cluster number, while the rest of the cells show the distance of the image object from the corresponding cluster center.

	1	2	…………	24
1	0.0057023	0.8752068	…………	1.1050291
2	0.001908	0.1215193	…………	1.0333817
. . .	. . .	. . .	…………	. . .
k	1.6914092	1.4451924	…………	1.0191518

Figure 8: Distance Matrix

The next job of the K-mean clustering process is to generate a Group matrix from these two center matrix and distance matrix. This is done by simply checking the minimum distance of an image object from the clusters. The object belongs to the cluster for which the distance from it is minimum. For each image object, a 0 or 1 is placed at each cell of the group matrix for showing its belonging to the cluster. Figure 9 demonstrate the group matrix formed, where the rows and the columns represents the same as the distance matrix.

	1	2	…………	24
1	1.0	0.0	…………	0.0
2	0.0	0.0	…………	0.0
. . .	. . .	. . .	…………	. . .
k	0/0	1.0	…………	0.0

Figure 9: Group Matrix showing image belongings

At this point when the belongings of all the image objects were found out, new centeroids were computed for each of the clusters. The new points of each cluster are now the mean value of the corresponding points of all the objects belonging to that cluster. Once the new center matrix was found, the whole process is repeated again at the next level of iteration for all the images. This results a new distance matrix and eventually a new group matrix.

This way the iterative process is continued until there is no further change in the group matrix. The K-mean clustering process stops when a final belonging of the images objects to the clusters is achieved.

Finally, data are extracted from the last group matrix and a resulting output of the training session is shown both textually as a reporting file and also visually as image files.

4.2.1 Results and Discussion

The output text file for 24 images with 10 clusters (i.e k = 10, supplied) is shown below.

```
User Inputs :

Number  of  Images:  24

Number  of  Clusters :  10

Number  of  Iterations :  1000


Clustering Results :

Cluster 1 Contains Objects :        1    18

Cluster 2 Contains Objects :        2    13

Cluster 3 Contains Objects :        3    15

Cluster 4 Contains Objects :        4    17    21

Cluster 5 Contains Objects :        5    14    19    20

Cluster 6 Contains Objects :        6    11    16

Cluster 7 Contains Objects :        7    12

Cluster 8 Contains Objects :        8    22

Cluster 9 Contains Objects :        9    24

Cluster 10 Contains Objects :      10    23


Total Execution time :  2 hours, 4 minutes, & 10 seconds.
```

Figure 10: Output File of The Training Session

Only two of the images (image 11 & 19) were misplaced at this particular experiment, indicating an achievement of 92% (approx.) accuracy by the training session. Each cluster represents a unique sign with a unique meaning. The maximum number of iterations to be performed was also supplied by the user, assuming that stability would be reached at an iteration level below this number. Time taken for the training phase of the system is also shown. In this case this time was 2 hours and 4 minutes approximately, which is quite a long duration. But as the training session is only performed once in order to load the system for a particular domain, this amount of time can be accepted.

The other output of the training session is a database containing the data about the clusters. This database is pretty much the same as the center matrix shown earlier. This database is one of the major inputs for the next classification phase.

4.3 CLASSIFICATION PROCESS

Once the system is loaded, the classification phase takes the control of the system. It is a continuous process, which takes random images as input and classifies them according to the clusters created by the training session. The classification process takes only a few seconds to classify given images and identify the meanings of the sign each of them represents.

The first part of the classification process is the same image processing as mentioned previously. The second part is the Neural Networking sub-process which actually classifies the images [25]. The Neural Network takes two input data – the data of the image objects from the output of the image processing and the data regarding the clusters from the database, produced by the training phase.

The Neural Networking is a supervised learning process. The system was designed to feed necessary data to this process dynamically. So the mapping classes and thus the number of neurons to be used by the Neural Networking are extracted from the output database of the training session. This is why the internal code or the architecture of the system needs not to be changed for a different domain of signs.

The Neural Networking process designed for this system is as shown in figure 11.

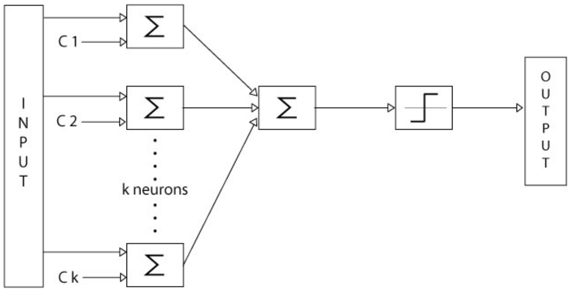

Figure 11: Neural Networking Process for each of the input images

In the following experiment, the number of clusters, k was equal to 10 and 12 new images (named 1001 to 1012) were introduced to the system. So, for each of the 12 input images 10 neurons were first used to find their corresponding distances from each of the 10 cluster centers. The distances were passed into another neuron which determines the minimum distance and thus identifies, to which cluster the image belongs to. This way the classification process completes when a group matrix was found for all the input images. The result is then shown on a text file.

Based on the resulting database of the training session, the efficiency of the Neural Network depends. If the database created was assumed to be 100% correct then the classification phase gives a 99% correct results.

5. LIMITATIONS AND FUTURE WORK

Due to the limited time constraint the system could not be developed to give its optimum performance. A lot of effort has to be done to make this intelligent system more effective.

One important work to be done in future is the code optimization of the current programs written for the system. This would certainly decrease the time complexity of the system to a very efficient level. Currently the system is based on the command prompt interface which would be converted to a graphical user interface soon.

At a very near future the classification part of the system would be extended, so that the system would be able to communicate in both directions. Depending on the same training session, the other classification part to be developed will have the capability to translate normal languages to hand gestures successfully. Both audio and video tools would be included with the input/output phase of the system to give it higher flexibility. This would enable the system to work equally effectively with vocal language and continuous signs. The image processing part of the system will also be modified to work with every possible environment.

Finally, the whole system needs to be compacted to act as complete stand alone software. It would then be handed over to a hearing impaired person. Performance evaluation, further research and fixing would be done on the system according to the comments of that person.

5. CONCLUSION

Although developing an automated system for dynamic sign language detection, is an extremely challenging job, but the importance of the necessity of such a system in the deaf community is comparatively huge.

So the basic objective of this research project was to develop an efficiently working methodology for sign language recognition system using modern technological concepts. The successful demonstration of this “Intelligent System” supports the idea and proves the efficiency of the explained methodology. The findings from this project created motivations to carry out further research in order to develop enhanced version of the proposed system. The hope was to influence the software development industry to research more on this particular field. So that, they can come up with an efficient and cost effective commercial product for the deaf society and help them carry out a normal life style.

REFERENCES

[1] Armstrong, D.F., Stoke, W.C., Wilcox, S.E., “Gesture and nature of language”, Cambridge Academic Press, 1995.

[2] Kadous, M.W., “Auslan Sign Recognition Using Computers and Gloves”. Available at : http://www.cse.unsw.edu.au/~waled.

[3] D. Shahriar Hossain Pavel, Tanvir Mustafiz, Asif Iqbal Sarkar, M. Rokonuzzaman, “Geometrical Model Based Hand Gesture Recognition for Interpreting Bengali Sign Language Using Computer Vision”, ICCIT, 2003.

[4] Adnan Eshaque, Tarek Hamid, Shamima Rahman, M. Rokonuzzaman, "A Novel Concept of 3D Animation Based 'Intelligent Assistant' for Deaf People: for Understanding Bengali Expressions", ICCIT, 2002.

[5] Khan Abul Bashar, Islam Santhe, Shaifulla Adnan Ifne, Shamrat Md. Anamul Haque, “Bengali Sign Language Recognition by Using Computer vision and 3D interactive Graphical Representation Tool for The Hearing disable People”, Thesis Paper, American International University – Bangladesh (AIUB)

[6] Frank W. Lovejoy Symposium, Co-Sponsored by Rochester Institute of Technology and the University of Rochester, "Application of Automatic Speech Recognition with Deaf and Hard of Hearing People", Rochester, New York, April 10-11, 1997.

[7] Sohalia Rahman, Naureen Fatema, M. Rokonuzzaman, "Intelligent Assistants for Speech Impaired People", ICCIT, 2002.

[8] Naureen Fatema, Towheed Chowdhury, M. Rokonuzzaman, “Vision based Dynamic Sign Language Recognition Using State Machines”, ICCIT, 2003.

[9] Sandberg, “Gesture Recognition using Neural Networks”, 1995.

[10] K. Murakami and H. Taguchi, “Gesture Recognition using recurrent neural networks”, in proceedings of CH191 Human factors in Computing Systems, 1991.

[11] Daw-Tung Lin, “Spatio-temporal hand gesture recognition using Neural Networks”, proceedings 1998 IEEE International Joint Conference on Computational Intelligence, Volume 3.

[12] Thad Starner, "Visual Recognition of American Sign Language using Hidden Markov Models" Master’s Thesis, MIT Media Lab, 1995. Available at: http://whitechapel.media.mit.edu/tech-reports/TR-316.ps.Z

[13] Anderberg, M.R., “Cluster Analysis for Applications”, New York: Academic Press, Inc., 1973.

[14] Hartigan, J.A., "Distribution Problems in Clustering," in Classification and Clustering, ed. J. Van Ryzin, New York: Academic Press, Inc., 1977.

[15] Wong, M.A., "A Hybrid Clustering Method for Identifying High-Density Clusters," Journal of the American Statistical Association, 77, 841-847, 1982.

[16] Hawkins, D.M., Muller, M.W., and Ten Krooden, J.A., "Cluster Analysis," in Topics in Applied Multivariate Analysis, ed. D.M., Hawkins, Cambridge: Cambridge University Press, 1982.

[17] Sergios Theodoridis, Konstantinos Koutroumbas, “ Pattern Recognition”, Elsevier Academic Press, 2003.

[18] Pollard, D., "Strong Consistency of k-Means Clustering," Annals of Statistics, 9, 135-140. 1981.

[19] Wong, M.A. and Lane, T., "A kth Nearest Neighbor Clustering Procedure," Journal of the Royal Statistical Society, Series B, 45, 362-368. 1983.

[20] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, and Angela Y. Wu, “An Efficient k-Means Clustering Algorithm: Analysis and Implementation”, IEEE Transaction on pattern analysis and Machine Intelligence, v.24, 2002.

[21] Frank Höppner, Frank Klawonn, Rudolf Kruse, Thomas Runkler, “Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition”.

[22] Frank Höppner, Frank Klawonn, “What is Fuzzy about Fuzzy Clustering ?”

[23] Ward, J.H., "Hierarchical Grouping to Optimize an Objective Function," Journal of the American Statistical Association, 58, 236-244. 1963.

[24] Kardi Teknomo, “K-mean Clustering Tutorial”, http://people.revoledu.com/kardi/index.html

[25] Atiqul Islam, Shamim Akhter, and Tumnun E. Mursalin. “Automated Textile Defect Recognition System Using Computer Vision and Artificial Neural Networks”, Transactions on Engineering, Computing and Technology, 2006.