Hand Gesture Recognition on Indian Sign Language using Neural Network

Deepali Kaushik
Department of Computer Science & Engineering
Krishna Institute of Engineering & Technology, Ghaziabad, U.P., India

Ankur Bhardwaj
Department of Computer Science & Engineering
Krishna Institute of Engineering & Technology, Ghaziabad, U.P., India

Abstract- This paper presents a novel technique for recognize the hand gesture during human–computer interaction. Frequently Hand gesture recognition system is designed such that no require any special hardware other than webcam. Human-Computer Interaction (HCI) is the study of interaction between users and computers. Our main objective is to explore the neural network-based approach to the recognition of the hand gestures. We have been used orientation histogram algorithm that will recognize hand gesture namely a subset of ISL (Indian sign language). Pattern recognition system used for transformation that convert an image into feature vector then compared to feature vector of training set of gestures. At last the final system is implemented with a perceptron network.

Keywords: HCI, ISL, ANN, Pattern Recognition.

I. INTRODUCTION

The idea is to make computers understand human language and develop a user friendly human computer interfaces (HCI). Making a computer appreciate speech, facial expressions and human gestures are some steps towards it [1]. Human-Computer Interaction (HCI) is the study of interface between users and computers [2].

Gestures are the non-verbally substitute information out of source of information. As we know a person can perform innumerable gestures at a time. A Gesture is a movement of the body that expresses an idea or attitude [3]. Hand gestures are a collection of movements of the hand and arm that vary from the static state of pointing at something to the dynamic state [4]. Now a day's hand gesture recognition became a famous topic in several area whether is computer society or electronics. Hand gestures recognition system has been applied for different applications on different domains including sign language, virtual environments, smart surveillance, robot control, medical systems etc [5]. In sign language, each gesture has an assigned meaning.

II. LITERATURE SURVEY

We have considered several previous works done in this field. This different works developed by different researchers. There are several approaches that were followed by different researchers like vision based, data glove based, Artificial Neural Network, Fuzzy Logic, Genetic Algorithm, Hidden Markov Model, Support Vector Machines etc. Some of the previous works are given below. . A fair amount of research has been done on different aspects of this approach.

In 1995 Freeman and Roth [13] both together introduced a method to recognize hand gestures, based on a pattern recognition technique developed by McConnell employing histograms of local orientation. Their method uses the orientation histogram as a feature vector for gesture classification. This method is fast and simple to compute and provide illumination changes in image.

In 1998 Naidoo et al. [14] is a member of another group of scholars who have equally suggested a system that recognizes static hand images against complex backgrounds. In this method, a Support Vector recognition system is

used to classify hand postures as gestures. Support Vector Machine (SVM) is a linear machine with some very good properties.

Another hand gesture recognition system has also been developed by Licsar and Sziranyi [15] in (2002) based on the shape analysis of static gestures for HCI. This appearance-based recognition system uses MFD (Modified Fourier Descriptors) for the classification of hand shapes.

A new approach has been presented by Chang et al in (2006) [16] which recognize static gestures based on Zernike Moments (ZMs) and Pseudo-Zernike Moments (PZMs). This approach includes four stages. In the first step, an input static gesture is segmented into a binary hand outline via the adapted color segmentation approach. In the second step, the binary hand outline is optional with a Minimum Bounding Circle (MBC).

Parvini and Shahabi in (2007) [17] have equally proposed an approach recognizing static and dynamic hand gestures by analyzing the raw streams generated by the sensors attached to human hands. The recognition of a sign is based on the observation of all forms of hand signs and finger-joint movements from a starting posture to a final posture.

D. Deora; N. Bajaj in 2012 'Indian sign language recognition' [18] presents a framework for a human computer interface capable of recognizing gestures from the Indian sign language. In this paper both Alphabets and numbers have been recognized successfully. This system can be extended for words and sentences. Here is PCA used for recognition. This paper also proposes recognition with neural networks.

V. Adithya; P. R. Vinod; U. Gopalakrishnan Artificial neural network based method for Indian sign language recognition [19]in 2013. They prepared a system that automatically recognizes the sign language. The implementation of such a system provides a platform for the interaction of hearing disabled people with the rest of the world without an interpreter. In this paper, we propose a method for the automatic recognition of finger spelling in Indian sign language. The proposed method uses digital image processing techniques and artificial neural network for recognizing different signs.

S. N. Sawant; M. S. Kumbhar. Real time Sign Language Recognition using PCA [20] .This paper presents the Sign Language Recognition system capable of recognizing different gestures from the Indian Sign Language by using MATLAB. The proposed system having four modules such as: pre-processing and hand segmentation, feature extraction, sign recognition and sign to text and voice conversion. Different features are extracted such as Eigen values and Eigen vectors which are used in recognition. The Principle Component Analysis (PCA) algorithm was used for gesture recognition and recognized gesture is converted into text and voice format. The proposed system helps to minimize communication barrier between deaf-dumb people and normal people.

P. V. V. Kishore; M. V. D. Prasad; C. R. Prasad; R. Rahul, 4-Camera model for sign language recognition using elliptical fourier descriptors and ANN [21] in 2015 Sign language recognition (SLR) is considered a multidisciplinary research area engulfing image processing, pattern recognition and artificial intelligence. The major hurdle for a SLR is the occlusions of one hand on another. This results in poor segmentations and hence the feature vector generated result in erroneous classifications of signs resulting in deprived recognition rate. To overcome this difficulty we propose in this paper a 4 camera model for recognizing gestures of Indian sign language. Segmentation for hand extraction, shape feature extraction with elliptical Fourier descriptors and pattern classification using artificial neural networks with backpropagation training algorithm. The classification rate is computed and which provides experimental evidence that 4 camera model outperforms single camera model.

III. APPLICATION OF HCI

As we know that vision based recognition is more utilizable recognition in this time and vision-based hand gesture recognition lately became a highly active research area. It motivated applications such as sign language recognition, socially assistive robotics, directional indication through pointing, control through facial gestures, human computer interaction (HCI), immersive game technology, virtual controllers, affective computing and remote control, smart surveillance, robot control, medical systems etc [5]. Overview of some hand gesture application areas are listed below

- Sign Language Recognition

- Robot Control
- Graphic Editor Control
- Virtual Environments (VEs)
- Numbers Recognition
- Television Control
- 3D Modeling

IV. INDIAN SIGN LANGUAGE

Indian Sign Language (ISL) is a well-known sign language, which is more use in different type of gesture. Now a day's ISL is used to recognize the action and especially it is used by deaf and dumb people [6]. ISL used to help the Indian hearing impaired people to interact with usual people with the help of computer. ISL are known as Deaf and Dumb languages. ISL is said to be gestural language [6]. ISL does not need speech channel for communication. There are mainly three approaches in the hand gesture analysis: glove based analysis, vision based analysis, and drawing gestures analysis.

It is generally categorized into 2 states.

1) Instruction manual (hand outline, hand position, direction and movements)

2) Non-manual (facial expression, eye stare and head/body pose).

In ISL, there can be one handed and two handed signs which can be static and dynamic state.

Before start to design of any sign language study system, it is an appropriate to use the chart of respective sign language hierarchy. Figure 1 shows ISL alphabet set.

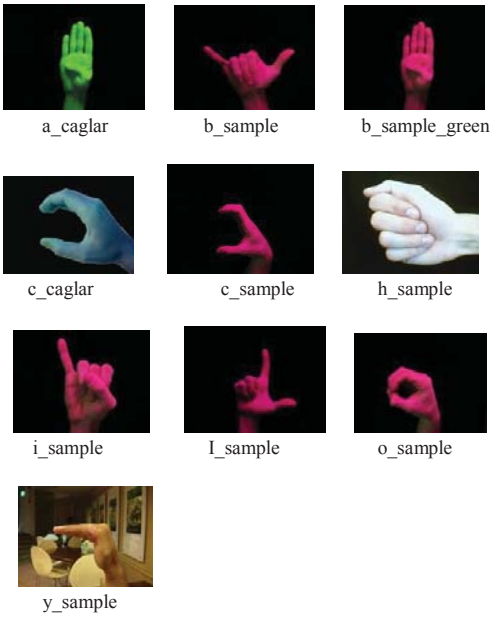

Fig-1 (ISL Alphabet Set)

V. NEURAL NETWORK

A neural network is a mainly parallel distributed processor equipped of simple processing units, which has a natural tendency for storing investigational knowledge and gathering it available for use. We can guide a neural network to

execute a particular function by modifiable the values of the connections (weights) between elements. Generally neural networks are regulated, or instructed, so that exacting input leads to a specific target output. Such a state is shown in Fig 2. Therefore the network is regulated, which is based on a comparison of the output and the target, until the network output matches the target [11].

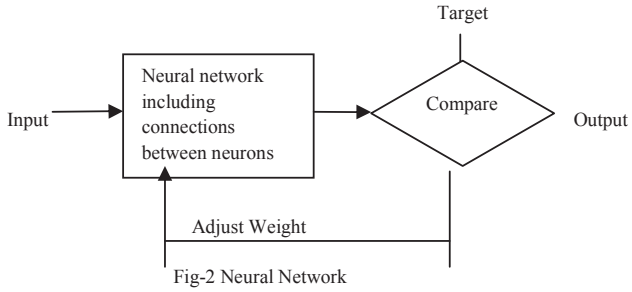

Fig-2 Neural Network

Neural networks have been trained to perform composite functions in a range of fields of application including pattern recognition, identification, classification, speech, and vision and control systems.

Supervised learning

Supervised learning is completely based on the training method which is trying to predict correct conclusions for known examples. It matches up to its predictions consequences to the target answer and "learns" from its mistakes. The data start as inputs to the input layer neurons. Initially the neurons pass the inputs down to the next nodes. As inputs are passed next to, the weighting, or connection is applied and when the inputs reach the next node, the weightings are summed and either builds up or damaged. This continues until the data get to the output layer where the model calculates an outcome. In a supervised learning classification, the predicted output is compared to the actual output. If the predicted output is equal to the actual output, no alter is made to the weights in the system. But, if the predicted output is higher or lower than the actual predicted outcome in the data, the error is propagated back throughout the system and the weights are adjusted consequently. This feeding error backwards through the network is known as "back-propagation". Multi-Layer Perceptron and the Radial Basis Function both are used as supervised learning techniques.

Unsupervised learning

Neural networks which use unsupervised learning are the largest part for describing data rather than predicting it. The neural network is not illustrating any outputs or answers as part of the training process--in reality, there is no perception of output fields in this type of system. The major unsupervised technique uses the Kohonen network to resolve the problem. The main uses of Kohonen and other unsupervised neural systems are in cluster analysis where the purpose is to group "like" cases together. The advantage of the neural network for this type of analysis is that it requires no initial hypothesis about what constitutes a group or how many groups there are. The system starts with a new slate and is not biased about which factors should be most important [10].

The Perceptron Convergence Algorithm

For the improvement of the error-correction learning algorithm for a single-layer perceptron, we will work with the signal-flow graph shown in Fig 3. In this method, the threshold θ (n) is treated as a synaptic weight connected to a fixed input equal to –1 [11].

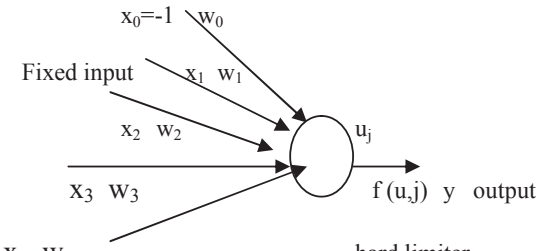

Fig- 3 Perceptron single flow graph

According to perceptron single flow graph, we may classify the (p+1) by-1 input vector as-

$$x(n)=[-1,x_1(n),x_2(n),\ldots\ldots\ldots..x_p(n)]^T$$

Correspondingly we identify the (p+1)-by-1 weight vector

$$w(n)=[\ \Theta(n),w_1(n),w_2(n)\ldots\ldots\ldots\ldots w_p(n)]^T$$

Below are some variable and parameters used in the convergence algorithm

x(n)= input

w(n)= weight

Θ(n)= threshold

y(n)=actual response

d(n)=desired response

η=learning rate parameter, $0< \eta<1$

So consent to the 4-step algorithm in detail:

Step 1: Initialization

Put w(0)=0. Then execute the following computations for time n=1, 2 . . . . .

Step 2: Activation

At time n, activate the perceptron by applying continuous-valued input vector x(n) and desired response d(n).

Step 3: calculation of Actual Response

Compute the actual response of the perceptron:

$$y(n)= sgn[w^T(n)x(n)]$$

The linear output is inscribed in the form:

$$u(n)= w^T(n)x(n)$$

where

$$\begin{cases} sgn(u) = +1 & \text{if } u>0 \\ sgn(u) = -1 & \text{if } u<0 \end{cases}$$

Step 4: variation of Weight Vector

$$w(n+1)= w(n)+ \eta[d(n)-y(n)]x(n)$$

where

$$\begin{cases} d(n) = +1 \text{ if } x(n) \text{ belong to class } C_1 \\ d(n) = -1 \text{ if } x(n) \text{ belong to class } C_2 \end{cases}$$

Step 5: Increment time n by 1 unit and go back to step 2.

VI. PROPOSED METHOD

Orientation Histograms

In orientation histogram approach we will found feature vector to classify the image. It can compute the local orientation using image gradients. I used two valve x and y derivative filters. The outputs of the x and y derivative

filter will be dx and dy. Then the gradient direction is at an (dx, dy). That's why I decided that we will use the edge orientation for a feature that will be show to the neural network. If edge detector is good then it will help to test the network with image database.
Figure 4 shows the orientation histogram for a simple image. Blurring histogram can be used to allow neighboring orientations to intellect each other.

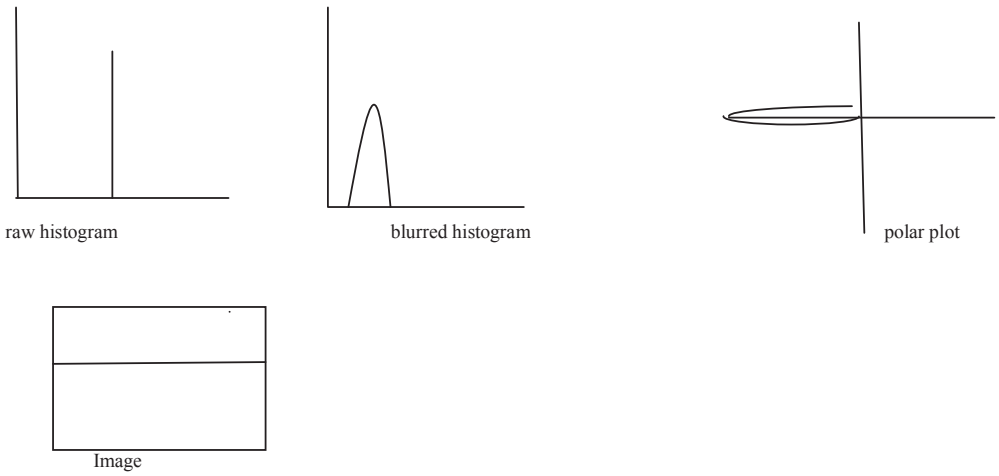

Fig-4 Orientation histogram

Function

Whole project can be 'divided' in 6 steps. We can see in flow chart fig-5. Let's observe them separately.

Step1

Firstly the program is coded as read the image database. There is for loop used to read a complete folder of image and store in MATLAB's memory. The folder is chooses by the user from menus. A menu will firstly come up inquiring you whether you would like to run the algorithm on test or train sets. Then second menus will come up for the user to decide which ISL sign wants to use. According to this situation we have to select any 1 sign for executing the program.

Step2

After that we have to resize all the images to read in pixels. This size appears the best for offering sufficient detail and it take low time in processing.

Step3

Next thing we will find the edges. To find the edges there would be 2 filter used.
For the x direction x = [0 -1 1]

For the y direction $y = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}$ which is the same as x but transposed and multiplied by –1.

This is used to only feature extracted from the images. I have also experimented with other known edge detectors as Sobel, Prewitt, Roberts and Canny. There is an appealing way of testing different edge detectors and altering their values is an Adobe Photoshop filter. Even some combinations of Canny or Sobel with line detection algorithms produced a very high-quality result.

Step 4

After that we have divided the two resulting matrices images dx and dy, element by element and then taking the atan ($\tan^{-1}$). This divider method will give the gradient orientation.

Step 5

After 4 steps we will use MATLAB function im2col which is used to rearrange the image blocks into columns. If we want to show the orientation histogram then step 5 is very necessary otherwise we can left this step. Developing algorithm histograms are the fastest way of getting a good detection.

Step 6

Now convert the column matrix with radian values to degrees. This way we can inspect the vector for values ranging from 0 to 90. It have done because for real elements of X, where atan(X) is in the range. Then determining the number of the histogram bins that is used to calculate experimenting values. Finally, the actual resolution of each bin was position to 10, which means 19 bins.

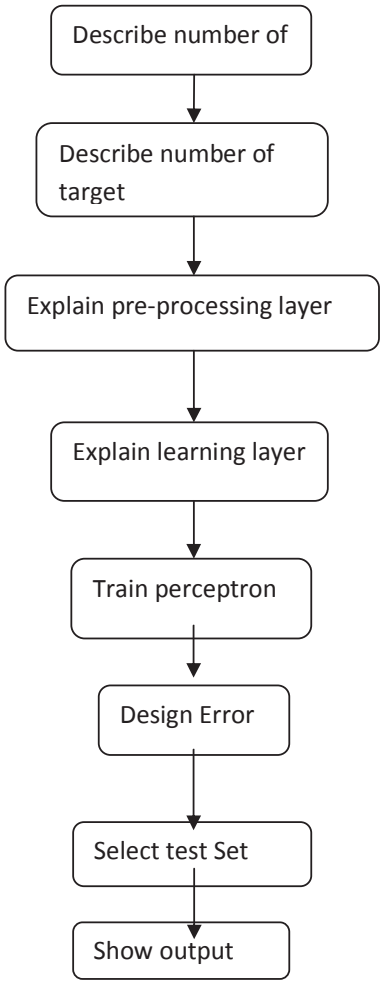

Fig 5 Perceptron flow- chart

VII. EXPERIMENT AND RESULTS

Following Experiments are performed using MATLAB R2012a and ADOBE Photoshop 7.0. Displaying the results in Fig- 7, Fig-8, Fig- 9, Fig-10, and Fig-11 acquired from the program. Different Sign images are considers which show different result. Firstly a pre-processing operation applied on different image and taken the parameter of blurring, noise as shown in the table 1. The table 1 provides us information about the pre-processing operations (i.e. blurring, noise). This operation is executed in Adobe Photoshop. If the image has noise then we try to extract it and find the matched database image from the input image. The result also fluctuates for every time when the algorithm is executed. The difference is very small. All the filtering operations were executed in Adobe Photoshop 7.0.

Blur – Calculated in pixel radius

Motion Blur – Similar to taking photo of moving car.

Noise - Either Uniform or Gaussian.

Now to recognize the input image from the database image we will execute the algorithm in MATLAB. Here are some images taken from database as shown below.

Fig-2 Fig-3 Fig-8

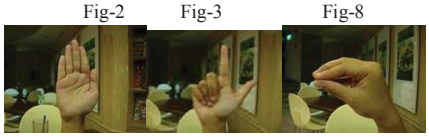

Fig-9 Fig-12 Fig-15

Fig-25

Fig- 6.1 Train Image

Now we will provide the input images and try to match from the databases images. As we will get the match image from the database image then the result is declared as match database image versus input image.

a_caglar b_sample b_sample_green

c_caglar c_sample h_sample

i_sample I_sample o_sample y_sample

Fig- 6.2 Test image

Now we will take a set of image in group as-
refresh gesture= y_sample
open gesture-= a_caglar, b_sample_green
close gesture-= h_sample
cut gesture= b_sample, i_sample, I_sample
copy gesture= c_caglar, c_sample, o_sample
Refresh gesture

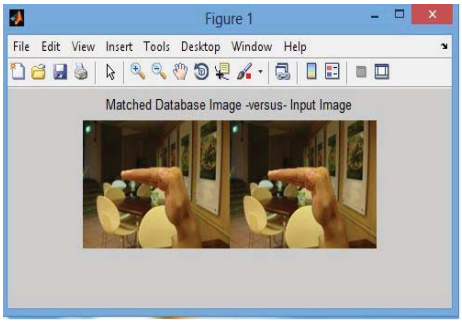

Fig- 7 Output Image-Refresh gesture

Open gesture

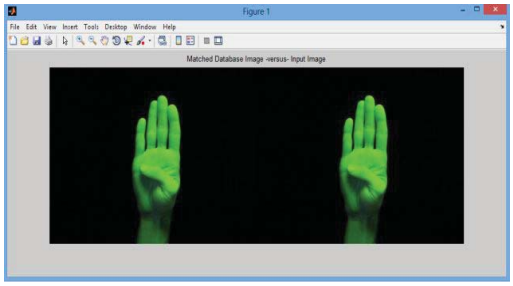

Fig- 8 Output Image-Open gesture

Close gesture

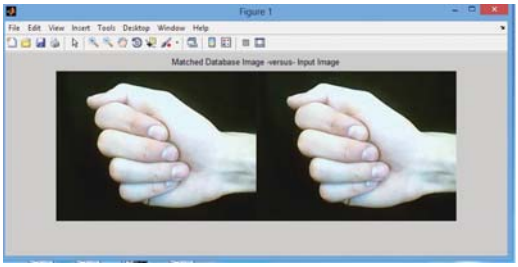

Fig- 9 Output Image- close gesture

Cut gesture

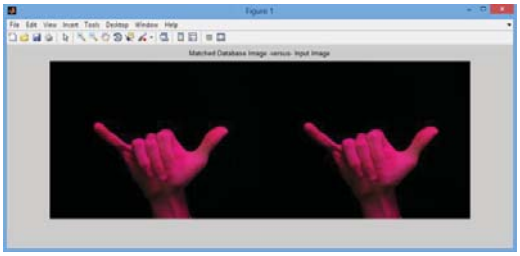

Fig- 10 Output Image- Cut gesture

Copy gesture

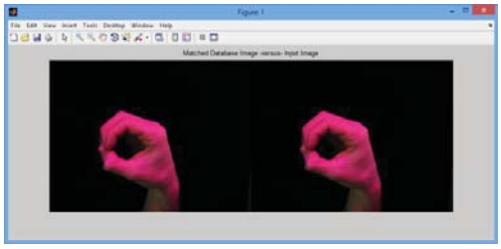

Fig- 11 Output Image- Copy gesture

Each gesture has a table of recognition results and with neural network outputs for one gesture image the performance of the proposed system is evaluated based on its ability to correctly recognized gestures to their corresponding input gestures, the metric that is used to accomplish this job is called the recognition rate. The recognition rate is defined as-

$$\text{Recognition rate} = \frac{\text{no of successful recognition gestures}}{\text{total no of gestures}} * 100\ \%$$

Table 1: Experiment Result

Gesture	Number of gesture	Successful recognition	Recognition rate %
Open	2	2	100
Close	1	1	100
Refresh	1	1	100
Cut	3	3	100
Copy	3	2	66.6
Total	10	8	93.32

Table 2: Comparison between different existing Methodologies:

Approach	Proposed Method	Using scale Normalization [22]	Using Gesture Segmentation [23]	Using Shape Fitting Technique [24]
Gesture recognition	93.32%	84%	92.78%	91.6%

It is describes from the above table that the performance of this method is 93.32% produced by neural network.

VIII. CONCLUSION

This project is started with the combination of Neural Networks. As we defined the working of orientation histograms, which is essential for classification that's why we will used it. In additional approaches of pattern recognition that orientation histograms have been used special conduct of comparing and classifying were utilized. It is an efficient algorithm. Another advantage of using neural networks is that we can illustrate conclusions from the network output.

It is finally declare that there is a robust conclusion at the end of the project. This is possible only for the first part of the project. Regardless of how many times you run the program the output vector will always be the same. This is not the case with the perceptron. Apart from not being 100% stable there are so many parameters (e.g. number of layers, number of nodes) that one can participate with that finding the optimal settings is not that straight forward. As mentioned earlier it all comes down to the application.

REFERENCES

[1] Meena, S.,2011.A Study on hand Gesture Recognition. Doctoral dissertation, National Institute Of Technology, Rourkela.

[2] van den Bergh, M., 2010. Visual body pose analysis for human-computer interaction. Diss, Eidgenössische Technische Hochschule ETH Zürich, Nr. 18838.

[3] Hai, W., 2002. Gesture Recognition Using Principal Component Analysis, Multi-Scale Theory, and Hidden Markov Models. Doctoral dissertation, Dublin City University.

[4] Dardas, N. H. A. Q., 2012. Real-time Hand Gesture Detection and Recognition for Human Computer Interaction. Doctoral dissertation, University of Ottawa.

[5] Mitra, S., & Acharya, T., 2007. Gesture recognition: A survey. systems, Man and Cybernetics, Part C: Applications and reviews, IEEE Transactions on, Vol. *37*, Issue 3, pp. 311-324.

[6] Zeshan, U., 2001. 'A' level Introductory course in INDIAN SIGN LANGUAGE .Ali. Yavar Jung National Institute for Hearing Handicapped, Mumbai, Vol. 1, Issue 38.

[7] Ibraheem, N. A., Khan, R. Z., & Hasan, M. M., 2013. Comparative study of skin color based segmentation *(IJAIS)*, Vol. *5*, Issue 10.

[8] Professor Charles A. Bouman's lecture notes gives introduction to colour space and chromaticity diagram: http://cobweb.ecn.purdue.edu/~bouman/ee637/notes/pdf/ColourSpace.pdf.

[9] Hasan, M. M., & Mishra, P. K., 2010. HSV brightness factor matching for gesture recognition system. *International Journal of Image Processing (IJIP)*, Vol 4, Issue 5, pp. 456-467.

[10] Dougherty, E. R., 1992. An introduction to morphological image processing.Tutorial texts in optical engineering.

[11] Bishop, C. M., 1995. *Neural networks for pattern recognition*. Oxford university press.

[12] Haykin, S., & Network, N., 2004. *A comprehensive foundation. Neural Networks*.

[13] Freeman, W. T., & Roth, M., 1995, June. Orientation histograms for hand gesture recognition. In *International workshop on automatic face and gesture recognition*, Vol. 12, pp. 296-301.

[14] Naidoo S, Omlin CW, Glaser M., 1998. Vision-based static hand gesture recognition using support vector machines. University of Western Cape, Bellville.

[15] Licsár A, Szirányi T., 2002. Hand-gesture based film restoration. Second international workshop on pattern recognition in information systems, Alicante, Spain, pp. 95–103.

[16] Chang CC, Chen JJ, Tai W, Han CC ., 2006. New approach for static gesture recognition. J Inform Sci Eng, Vol.22, pp.1047–1057.

[17] Parvini F, Shahabi C., 2007. An algorithmic approach for static and dynamic gesture recognition utilizing mechanical and bio. characteristics. Int J Bioinform Res Appl, Vol. 3, Issue 1.
[18] Deora, D., & Bajaj, N., 2012, December. Indian sign language recognition. In Emerging Technology Trends in Electronics, Communication and Networking ET2ECN, 1st International Conference on pp. 1-5.
[19] Adithya, V., Vinod, P. R., & Gopalakrishnan, U., 2013, April. Artificial neural network based method for Indian sign language recognition. In Information & Communication Technologies (ICT), IEEE Conference on pp. 1080-1085.
[20] Sawant, S. N., & Kumbhar, M. S. , 2014, May. Real time Sign Language Recognition using PCA. In Advanced Communication Control and Computing Technologies (ICACCCT), International Conference on (pp. 1412-1415).
[21] Kishore, P. V. V., Prasad, M. V. D., Prasad, C. R., & Rahul, R., 2015, January. 4-Camera model for sign language recognition using elliptical fourier descriptors and ANN. In Signal Processing And Communication Engineering Systems (SPACES), International Conference on pp. 34-38.
[22] M. M. Hasan and P. K. Mirsa, "Brightness factor matching for gesture recognition system using scaled normalization," International Journal of Computer Science & Information Technology, vol. 3, no. 2, 2011.
[23] V. S. Kulkarni and S. D. Lokhande, "Appearance based recognition of american sign language using gesture segmentation," International Journal on Computer Science and Engineering, vol. 2, no. 3, pp. 560-565, 2010.
[24] E. Stergiopoulou and N. Papamarkos, "Hand gesture recognition using a neural network shape fitting technique," Elsevier Engineering Applications of Artificial Intelligence, vol. 22, no. 8, pp. 1141-1158, 2009.