Middle-East Journal of Scientific Research 25 (7): 1407-1413, 2017
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2017.1407.1413

ASL Numerals Recognition from Depth Maps Using Artificial Neural Networks

M.V. Beena and M.N. Agnisarman Namboodiri

Vidya Academy of Science & Technology, Thrissur - 680501, India

Abstract: American Sign Language (ASL) is a sign language widely used by members of the Deaf community to communicate among themselves. Since people with normal hearing are not conversant with the sign language used by deaf-mute persons, a deaf-mute person generally finds it difficult to communicate with normal hearing people. This problem is the motivation behind the attempts to develop software for the machine recognition of the signs used in sign languages like ASL. Most of these are focused on successful recognition of a small subset of the the large number of signs used in any sign language. Along these lines, this paper presents a new algorithm for the machine recognition of the signs for numerals in ASL. The algorithm makes use of static depth images captured using Kinect sensors. The technique was executed using a large dataset having 10, 000 samples consisting of 1, 000 samples of each ASL numeral. The system extracted the features such as mean absolute value, frequency change, standard deviation, gradient change and zero crossing to train the neural network using SCG algorithm and translated different signs into text format. A total of 392 features were extracted using different feature extraction algorithms and achieved a maximum efficiency of 99.46%.

Key words: American Sign Language (ASL) · Feature extraction · Classification · ANN · SCG Algorithm · Depth map

INTRODUCTION

Sign language is the common way of communication medium for physically Impaired communities. As a kind of gesture, sign lan- guage is the primary communication media for deaf people. Gestures are used by deaf people to get useful information and exchange ideas. Therefore recognition of automatic sign language aims to understand the meaning of signs without the assistance from experts. Sign language translators are used for interpreting their thoughts between normal people. But it becomes very difficult to find a well educated and experienced translator for the conversion of sign language . So human-computer in- teraction system for this can be introduced anywhere possible. So a system recognizing gestures for sign language automatically is nec- essary which will help to ensure that deaf and dump people have equality of opportunity and full participation in the society. Recognizing Sign language is still a challenging despite of many research efforts for few decades. Depth images captured from Kinect sensor enable to capture additional information to improve accuracy and/or processing time. The recent ad- vancement of GPU, ANNs have been proposed to many computer vision problems. The reason for this is the reduced time for testing and training time when using GPU compared to CPU. Therefore this system took the advantage of depth sensor images and feed forward ANN to create an accurate sign language classifier for counting numbers.

Related Works: Many researchers have been formulating dif- ferent works on the sign language problem of using different kinds of methods and devices for decades. Recently some solution on ges- ture language identification using sensors and image processing methods have been imple- mented. The overall objective of these tech- niques is to help the communication of disabled people easy and replace traditional lan- guage by gesture language. Sign language uses many gestures so that it looks like a movement language which consists of series of hands and arms motion facial expression and head/body features. There are various standards like ISL, ASL, Arabic sign language, Australian sign lan- guage, BSL etc for sign languages. For recog- nizing gestures a series of feature extractions , such as SIFT method [1, 2], Histogram of Based Gradients [3, 4, 5], Wavelet Moments [6] and Gabor Filters [7, 8] were developed. Typical classifiers include Artificial Neural Networks, SVM, Decision Trees, etc. The above methods were recognized on a few hand gestures. Using SIFT algorithm and SVM

Corresponding Author: M.V. Beena, Vidya Academy of Science & Technology, Thrissur - 680501, India.

classifier [1] 96.23% efficiency was obtained for six hand gesture. Nagi *et al*. [10] presented system for inter- acting with human robots using CNNs. Van Bergh and and Van Gool. L [11] proposed a method by Haar wavelet features. Although these methods show better results, which con- sidered six classes of gestuers only. L. Pigou *et al*. [12] developed a CNN algorithm for recog- nizing italian sign language. S. Liwicki *et al*. [13] proposed a system for recognizing letters by Histogram of Gradients descriptors for British sign language. T. Starner *et al*. [14] proposed a system which recognizes a sentence of few words.

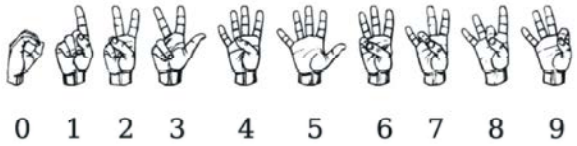

Fig. 1: ASL numerals ([21])

S. Foo *et al*. [15] proposed an ASL finger spelling recognition system using ANN and wavelet features. They have achieved an accu- racy rate for recognition as 91%. R. Bowden *et al*. [16] proposed an ASL finger spelling recog- nition system using random forest classifica- tion method in real-time. They recognized 24 alphabets of ASL finger spelling for alphabets. They have used dataset of five subjects and achieved recognition rate of 75% only. Very recently, M. Leu *et al*. [17] proposed an ASL alphabet recognition system. They first local- ized hand joint positions using random forest and hierarchical mode-seeking method. This system is recognized ASL signs by applying random forest classifier to joint angle vector. Their accuracy is 90% for trained images and 70% accuracy for new members.

Proposed Work: The proposed work is used to create a platform for sign language education for deaf and dumb people community. There are different methods for classification for recognizing different sign languages for HCI. The objective of this work is to develop a system that uses bare hand gestures for ASL numerals in the vision-based setup with maximum efficiency. The proposed architecture for the system is shown in Figure 2.

The proposed system consists of the follow- ing steps:

- The pre-processing steps are built on the basis of Histogram equalization and canny edge detection algorithms.

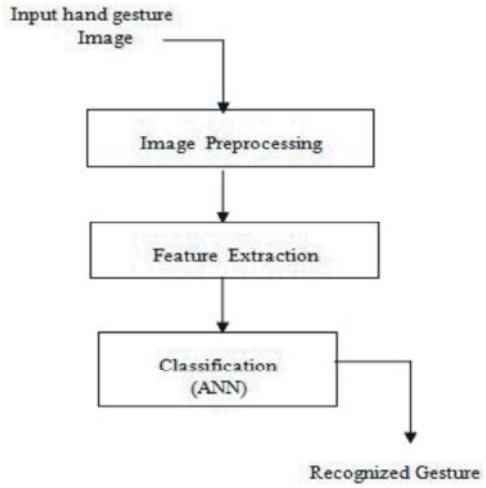

Fig. 2: Block diagram of gesture recognition

- One of the highlights of the system is the feature extraction using different al- gorithms which uses means and standard deviation of the pixel values from the block processed images.Total 392 features were extracted for classification purpose.
- Classification is done by using the SCG algorithm.

Application dependent features like finger- prints, faces and other conceptual features. In the proposed work the entire image block is split into number of blocks and features were extracted using mean and standard deviation.

Dataset: For this work, the publically available images using a depth sensor, creative senz3D camera of 320x240 have collected for 0 to 9 numbers. The system used data set of 10000 depth images for 10 different hand signs to represent numbers in ASL. The collected data set images modified according to our needs. Even though the down- loaded data set images reduced to 28x28 grey scale images, we achieved maximum accuracy without any loss of information.

Pre-processing is a process of preparing data used for another procedure. This pre- process- ing steps aims to transform those data into a form which can be more easily and effectively processed. In this work, the pre-processing steps are built on the basis of several combi- nations from the following image processing operations: gray scale image processing , edge detection, histogram equalization.

Histogram equalization is used in the first step of pre-processing of images. HE is the way of distribution of uniform grey scale values in an image. In this paper secondly we used edge detection algorithms

for pre-processing. Edges are having maximum role in object iden- tification and recognition, since they represent the boundary between an object and the back-ground or between adjacent objects. Therefore, extracted edge features are used in pattern recognition/ classification applications. In this work canny algorithm is used for edge de- tection. The traditional canny detector divided into four categories, Gaussian filter, Gradient Calculation, Non maximum suppression and hysteresis thresholding [19, 20]. For smoothen- ing the image and reduce the obvious noise, a Gaussian filter used to convolve with the image. Magnitude and direction of gradient is used to calculate non-maximum suppression and it is also used for thinning the edge of image.

After Non-maximum suppression, the edge extracted are almost the real edge, but certain edge cause by noise are still present. To remove these unwanted edges, two threshold value are empirically assigned such that if the edge pixel gradient is more than the threshold limit, it is taken as the strong edge. If the gradient value of edge pixel is lower than the threshold value, the edge pixel is suppressed. If the value of gradient of edge pixel is lower than the value of threshold, it is taken as weak edges and is compared with its neighbourhood pixel. If the pixel is connected with a strong edge then it is taken as a true edge.

Feature Extraction: The feature is defined as a function which may be one or more measurements, each of which specifies some quantifiable property of an ob- ject and is computed such that it quantifies some significant characteristics of the object.

The various feature classifications are cur- rently employed as follows.

- General features

Color, texture and shape are the appli- cation independent features. Abstraction level features are again classified into three categories:

- Pixel-level features: Features calcu- lated at each pixel, e.g. color, location.
- Local features: Features calculated over the results of subdivided image band on image segmentation or edge detection.
- Global features: Features calculated in the whole image or just regular block of an image.
- Domain-specific features

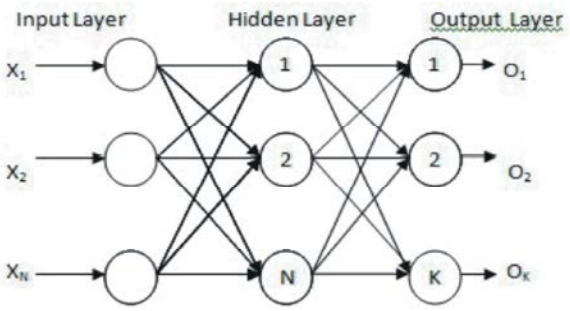

Fig. 3: Feed forward Neural Network

Application dependent features like fin- gerprints, faces and other conceptual fea- tures. In the proposed work the entire image block is split into number of blocks and features were extracted using mean and standard deviation.

Classification: Feature vector obtained from the feature extrac- tion step is used as the input of the classifier that recognizes the sign. The relavent and basic properties of the neural networks are training and generalizing [18]. Hence, ANN is used as the classification tool. Different network mod- els exist for training the neural net. Based on feature vectors, the best neural net training method has been chosen. An artificial neural network processes information by creating con- nections between artificial neurons and they are widely used to design complex relationship among outputs and inputs. Training or learning can configure a neural network for specific application to produce desired outputs. Various algorithms exist to train an artificial neural network. In feed forward neural network, each neuron obtain the signal from the preceding layer and another weight walue is multiplied to each signal.The sum of weighted inputs is fed to a scaling function which produces the output to a fixed limiting values . The archi- tecture for feed forward network is given in following Figure 3 having x as input and o as output response.

The Scaled Conjugate Gradient(SCG) method is used in this system for optimization purpose.

SCG collects second order data from neural network but it needs only O(N) in the case of memory, where N represents the weights in the network. The SCG performance is more than that of standard back propagation algorithm (BP), the conjugate gradient back propagation (CGB) and the one-step memory less quasi- Newton algorithm (BFGS). The proposed ar- chitecture and transfer function is depicted in Figure 4 and 5.

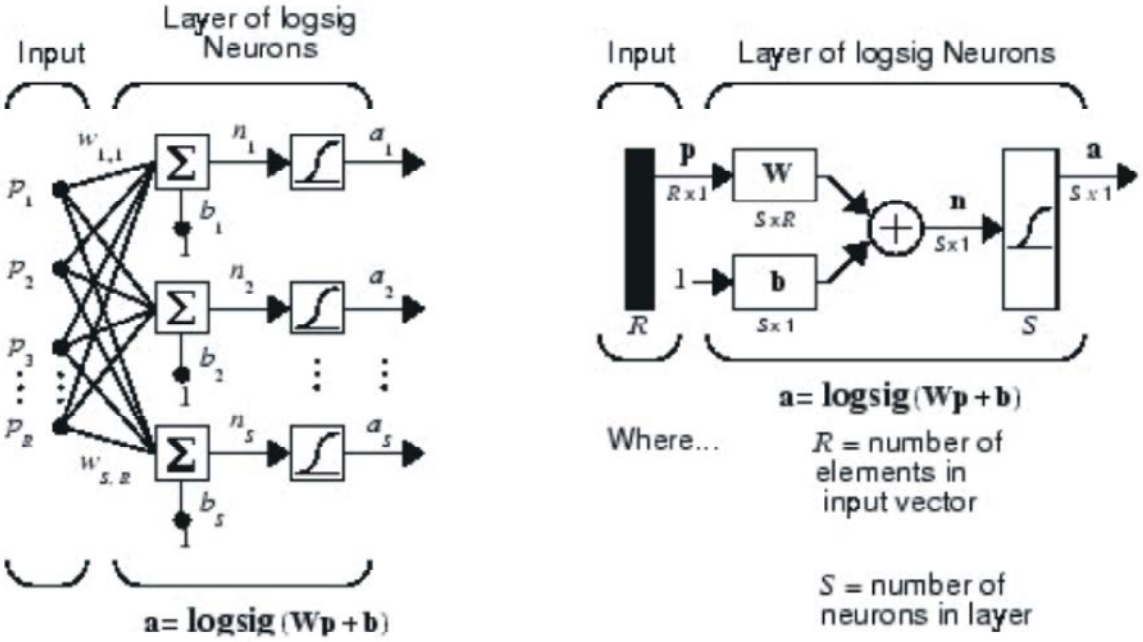

Fig. 4: Feed forward Neural Network using Log-Sigmoid Transfer Function

$$logsig(n) = 1 / (1 + exp(-n))$$

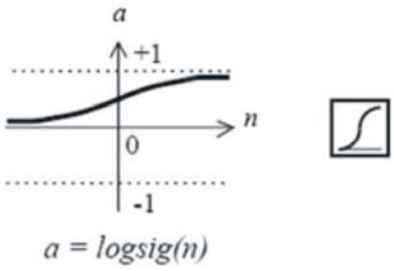

Fig. 5: Log-Sigmoid Transfer Function: Defini- tion, graph and symbol

The extracted 392 features of image are fed as input to the network. So, input layer con- tains 392 neurons. Output layer consists of ten neurons, as images are classified into ten classes, for static signs. Log sigmoid used as the transfer function .

Performance Parameters: The selection of correct metrics for evaluating the performance of system is vital to the result and the validation of the system. The parameters are selected by measuring the effective- ness of the processes involved in it. Sensitivity, specificity and accuracy are chosen to evalu-ate performance of the proposed method. The best training performance is achieved within 1 minute 30 sec for one epoch with 80 hidden layers using 10000 images. The performance is calculated for different hidden layers using NVIDIA Geforce GTX 940 M graphics card on Intel 6th generation i7 processor.

Table 1: Description of proposed neural network

Input neurons	392
Hidden neurons:	80
Output neurons:	10
Activation function:	SCG

Sensitivity: This measures the ratio of actual positives to the sum of true positives and false negatives:

$$Sensitivity = \frac{TP}{TP + FN'} \quad (1)$$

where TP is count of true positives (i.e. relevant items that are correctly identified as relevant) and FN is number of false negatives (i.e. rel- evant items that are incorrectly identified as irrelevant).

Specificity: It ratio of true negatives to the sum of true negatives and false positives.

$$Specificity = \frac{TN}{TN + FP} \quad (2)$$

Accuracy: It is the ratio of the test results that is true positive and true negatives to the sum of over all cases. The over all accuracy obtained is 99.46%.

$$Accuracy = \frac{TP + TN}{TN + FN + TP + FP} \tag{3}$$

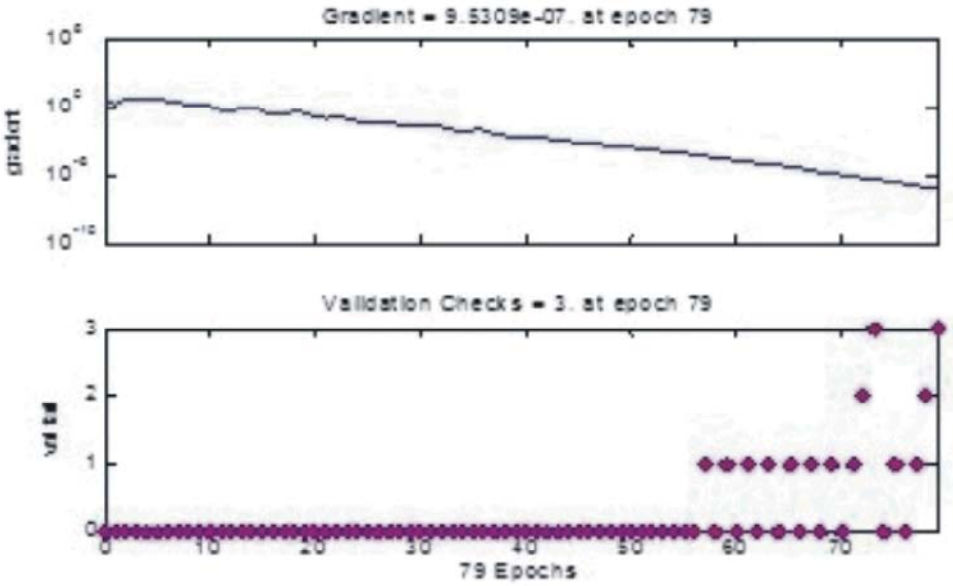

Fig. 6. Nueral network training state

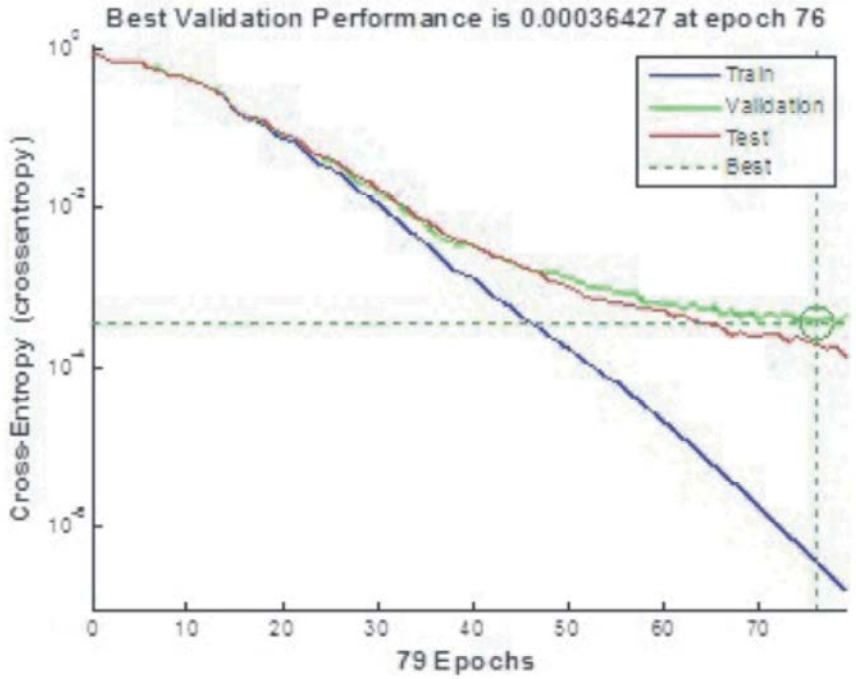

Fig. 7: Performance of neural network classifier

Experimental Results and Discussion: Implementation of system is carried out using MATLAB 2017a. The proposed methodology is tested for 0 to 9 numbers in ASL, for static signs. 90% of the data in dataset is used for training and the remaining 10% for validation and testing. The results for Static signs are shown in Table 2. It has been observed that the most of signs are correctly classified to their respective class.

Table 2: Performance Evaluation

	Rate				
Signs (class)	False negatives (FN)	False positives (FP)	True Positives (TP)	True negatives	Accuracy
0	0.0147	0	1	0.9853	0.9926
1	0	0.0083	0.9917	1	0.9958
2	0	0	1	1	1
3	0.0091	0	1	0.9909	0.9726
4	0	0	1	1	1
5	0	0	1	1	1
6	0	0.0061	0.9939	1	0.9969
7	0	0	1	1	1
8	0	0.0037	0.9963	1	0.9981
9	0.0019	0	1	0.9981	0.990

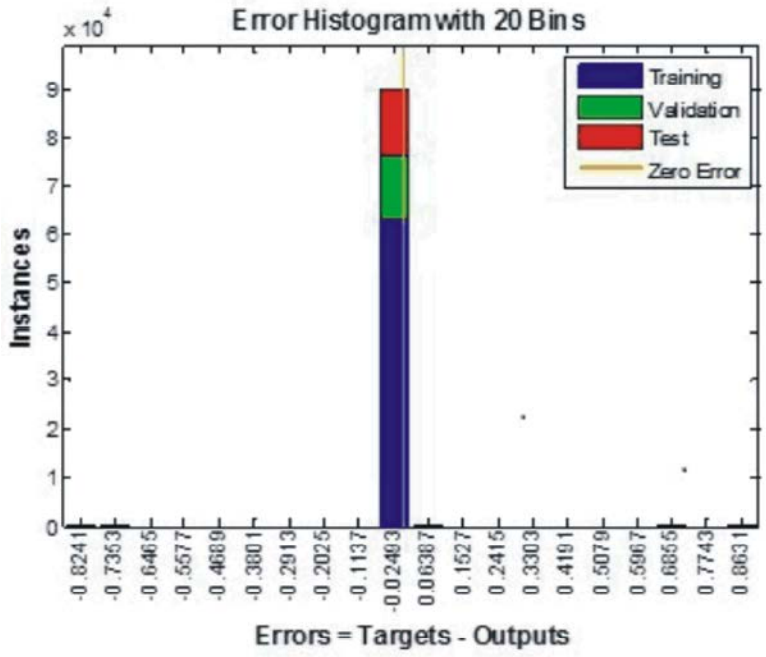

Fig. 8: Error histogram of neural network classi- fier

CONCLUSION

Communications between deaf mute and a nor- mal person has always been a challenging task. The goal of this project is to reduce barrier of communication by contributing to the field of automatic sign language recognition. Artifi-cial Neural Network based method for recog- nizing American Sign Language for numerals is implemented in this work with maximum recognition rate of 99.46% efficiency. It enables human beings to interact with machine in a more natural way. As the method implemented using machine learning and image processing techniques , the user does not have to wear any special hardware devices like sensors to recognize gestures.

REFERENCES

1. Dardas, N.H. and N.D. Georganas, 2011. Real-Time Hand Gesture Detection and Recognition Using Bag-of-Features and Support Vector Machine Techniques, Instrumentation and Measurement, 60: 3592–3607.
2. Gurjal, P. and K. Kunnur, 2012. Real Time Hand Gesture Recogni- tion Using SIFT, International Journal of Electronics and Electrical Engineering.
3. Nlker, C. and H. Ritter, Detection of Fingertips in Human Hand Movement Sequences, Gesture and Sign Language in Human-Computer Interaction, 1371: 209-218.

4. Nlker, C. and H. Ritter, 1999. GREFIT: Visual Recognition of Hand Postures, Gesture and Sign Language in Human- Computer Interaction, 1739: 61-72.
5. Mihalache, C.R. and B. Apstol, 2013. Hand pose estimation using HOG features from RGB-D data, System Theory, Control and Computing (ICSTCC), pp: 356-361.
6. Chen K., X. Guo and J. Wu, 2013. Gesture recognition system based on wavelet moment, Applied Mechanics and Materials, 401-403, pp: 1377-1380.
7. Amin, M.A. and H. Yan, 2007. Sign Language Finger Alphabet Recognition from Gabor-PCA Representation of Hand Gestures, Machine Learning and Cybernetics, 4: 2218-2223.
8. Pugeault, N. and R. Bowden, 2011. Spelling It Out: Real-Time ASL Fingerspelling Recognition, IEEE Workshop on Consumer Depth Cameras for Computer Vision.
9. Dardas, N.H. and N.D. Georganas, 2011. Real-Time Hand Gesture Detection and Recognition Using Bag-of-Features and Support Vector Machine Techniques, Instrumentation and Measurement, 60: 3592-3607.
10. Nagi, J., F. Ducatelle, G. Di Caro, D. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber and L. Gambardella, 2011. Max-pooling convolutional neural networks for vision- based hand gesture recognition. In Signal and Image Processing Applications (ICSIPA), 2011 IEEE International Conference on, pp: 342347, Nov 2011.
11. M. Van Den Bergh and L. Van Gool, 2011. Combining rgb and tof cameras for real-time 3d hand gesture interaction. In Appli- cations of Computer Vision (WACV), 2011 IEEE Workshop on, pp: 6672, Jan 2011.
12. Pigou, L., S. Dieleman, P.J. Kindermans and B. Schrauwen, 2015. Sign language recognition using convolu- tional neural net- works. In Computer Vision - ECCV 2014 Workshops, pp: 572578.
13. Liwicki, S. and M. Everingham, 2009. Automatic recognition of fingerspelled words in british sign language. In Computer Vi- sion and Pattern Recognition Workshops, 2009. CVPR Work- shops 2009. IEEE Computer Society Conference on, pp: 5057, June 2009.
14. Zafrulla, Z., H. Brashear, T. Starner, H. Hamilton and P. Presti, 2011. American sign language recognition with the kinect. In Proceedings of the 13th International Confer- ence on Multimodal Interfaces, ICMI 11, pp: 279286. ACM, 2011.
15. Isaacs, J. and S. Foo, 2004. Hand pose estimation for american sign language recognition. In System Theory, 2004. Pro- ceedings of the Thirty-Sixth Southeastern Symposium on, pp: 132-136.
16. Pugeault, N. and R. Bowden, 2011. Spelling it out: Real-time asl fingerspelling recognition. In Computer Vision Work- shops (ICCV Workshops), 2011 IEEE International Con- ference on, pp: 11141119, Nov 2011.
17. Dong, C., M. Leu and Z. Yin, 2015. American sign language alpha- bet recognition using microsoft kinect. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2015 IEEE Conference on, pp: 4452, June 2015.
18. Adithya, V., P.R. Vinod and Usha Gopalakrishnan, 2013. Artificial Neural Network Based Method for Indian Sign Language Recognition , IEEE Conference on Information and Com- munication Technologies (ICT), pp: 1080-1085.
19. Canny J., 1986. A computational approach to edge detection. Pattern analysis and machine intelligence. IEEE transac- tion on, PAMI, 8(6): 679-698.
20. Srinivas, B.L. and Hemalatha K.A. Jeeran, 2015. Edge detection tech- nique for Image segmentation. International Journal of Innovative Research in Computer and Communication Engineering, 3(Special Issue 7), 2015.
21. Wikimedia Commons, Accessed on June 8, 2017 from https://commons.wikimedia.org/wiki/File:Asl_alphabet_gallaudet.png.