Volume 4, Issue 12, December 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Comparison between Unsupervised Feature Learning Methods Using an Auto-Encoder and Self-Organizing Map Algorithm for Sign Language Recognition

Sweta Shiwani*
Department of Electronics & Communications Engineering,
Gyan Ganga College of Technology, Jabalpur,
RGTU, Bhopal, India

Neeraj Shukla
Department of Computer Science Engineering,
Gyan Ganga College of Technology, Jabalpur,
RGTU, Bhopal, India

Abhishek Kumar
Department of Computer Science Engineering
Satya Sai Institute of Technology, Sehore,
RGTU, Bhopal, India

Abstract— In this paper we compared two methods for Sign Language Recognition (SLR) of some alphabet, first method for unsupervised feature learning using a sparse autoencoder, followed by training as Softmax classifier. Softmax classifier again learns using the L-BFGS optimization function .Second method using a Self-Organizing map algorithm for classification of DCT based feature vectors. Self-Organizing Map (SOM) is one of the most popular neural network models.

Keywords—Unsupervised feature learning, Sparse Autoencoder, Softmax classifier, Self-organising map algorithm ,Discrete cosine transform

I. INTRODUCTION

Communication is the process of exchanging information, views and expressions between two or more persons, in both verbal and non-verbal manner. Hand gestures are the non-verbal method of communication used along with verbal communication. A more organized form of hand gesture communication is known as sign language.

The Sign language is very important for people who have hearing and speaking deficiency generally called Deaf and Mute. It is the only mode of communication for such people to convey their messages and it becomes very important for people to understand their language. This paper compares the method or algorithm for an application which would help in recognizing the different signs which is called Indian or American Sign Language. For comparison, we may use different algorithm like Unsupervised feature learning, Self-Organizing Map (SOM) etc.

For Unsupervised feature learning method we first take a data set showing different hand gesture of ISL or ASL alphabet that we wish to classify. Now use this hand gesture data for unsupervised feature learning using an auto encoder followed by a training of Softmax classifier for making a decision about which letter is being displayed.

In Self Organizing Map method, features are extracted from hand gesture images based on skin pixels through image compression using two dimensional Discrete Cosine transform. A Self Organizing Map (SOM) an unsupervised learning technique in artificial neural network is used for classification of DCT based feature vectors.

II. NOTEWORTHY CONTRIBUTION IN THE FIELD OF PROPOSED WORK

In this work, we will study different unsupervised learning methods as:

I. Sparse autoencoders,
II. Self-Organizing map (SOM)
III. 2D-Discrete Cosine transform
IV. Softmax classifier.

We briefly summarize how these algorithms are employed in our system:

I. Sparse Autoencoder: An sparse autoencoder is an artificial neural network used for learning efficient coding .It is an approach to automatically learn feature from unlabeled data .By training a neural network,it produce an output that is identical to the input but having fewer nodes in the hidden layers that is in input .we have built a tool for compressing the data.
II. Self-Organizing Map (SOM):- The Self-Organizing Map is one of the most popular neural network models. It belongs to the category of competitive learning networks. The Self-Organizing Map is based on unsupervised

learning, which means that no human intervention is needed during the learning and that little need to be known about the characteristics of the input data.

SOM has the capability to generalize. Generalization capability means that the network can recognize or characterize inputs it has never encountered before. A new input is assimilated with the map unit it is mapped too.

The Self-Organizing Map is a two-dimensional array of neurons.

III. 2D-Discrete Cosine Transform: The discrete cosine transform (DCT) is a technique for converting a signal into elementary frequency components. It is widely used in image compression. Here we develop some simple functions to compute the DCT and to compress images. These functions illustrate the power of Mathematical in the prototyping of image processing algorithms. For analysis of two-dimensional (2D) signals such as images, we need a 2D version of the DCT. For an n x m matrix s, the 2D DCT is computed in a simple way: The 1D DCT is applied to each row of s and then to each column of the result.

IV. Softmax classifier: - The Softmax algorithm is a good algorithm to use for the classification of basic problems that are linearly separable. The main advantage of the Softmax Classifier is that it consists of a very simple model and is therefore very fast to train and predict.

III. PROPOSED METHOD

The proposed method as described is based on following discussed techniques: Unsupervised Feature Learning, Autoencoder, Softmax classifier, L-BFGS optimization function. This method consists of two stages: Data collection and Hand Segmentation, Unsupervised Feature Learning and Classification.

Data collection and Hand Segmentation

Figures 1: Samples of each letter from our dataset

For data collection, we use a Microsoft Kinect 3D depth camera. Videos were taken of the test subject's hands while forming sign language letter. Frames showing individual letter were extracted. For segmenting out only the hand, we tried several approaches as described below:

The first approach involved modelling the skin colour by a 2D Gaussian curve and then using this fitted Gaussian to estimate the likelihood of a given colour pixel being skin. We first collected skin patches from 40 random images from the internet. Each skin patch was a contiguous rectangular skin area. The patches were collected from people belonging to different ethnicities so that our model is able to correctly predict skin areas for a wide variation of skin colour. We first normalized each colour as follows:

$r=R/(R+G+B)$, $b=B/(R+G+B)$.

We ignored the G component as it is linearly dependant on the other two. We then estimated the mean and covariance matrix of the 2D Gaussian (with r, b as the axes) as Mean $\mu = E(x)$, Covariance $C = E(x - \mu)(x - \mu)T$, where x is the matrix with each row being the r and b values of a pixel. With this Gaussian fitted skin colour model, we computed the likelihood of skin for any pixel of a given test image. Finally, we determine threshold the likelihood to classify it as skin or non-skin. However, this approach did not give significantly good results and failed to detect dimly illuminated parts of skin. The results of using this algorithm for skin segmentation are shown in figures below.

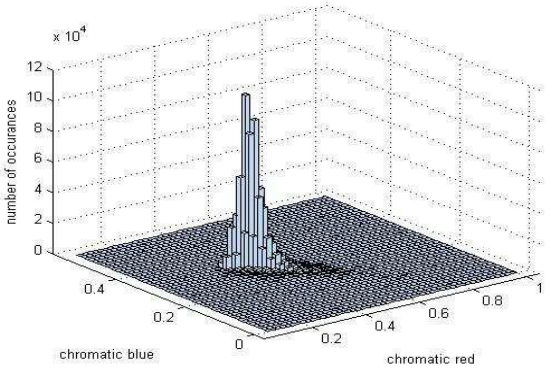


Fig.2(a): Histogram of the colour distribution for skin patches and the corresponding Gaussian model that was fit to it.

The second approach, which we used is motivated by the paper, in which we first transform the image from the RGB space to the YIQ and YUQ colour spaces.Then the compute the parameter $\Theta=\tan^{-1}(V/U)$ and combine it with the parameter I to define the region to which skin pixels belong. Specifically, we called all pixels with $30 < I < 100$ and $105^o < \Theta < 150^o$ as skin. For our experiments, we tweaked these thresholds a bit and found that the results were significantly better than our Gaussian model in the previous approach. This might have been because of two reasons – (i) our Gaussian

model was trained using data samples of insufficient variety and hence was inadequate to correctly detect skin pixels of darker shades (ii) fitting the model in the RGB space performs poorly as RGB doesn't capture the hue and saturation information of each pixel separately. After having detected the skin regions, the next task was to filter out only the hand region and eliminate the face and other background pixels that might have been detected. For this, we used the depth information that we obtained from the Kinect, which returns a grayscale depth image with objects having lower intensity values being closer to the camera. We assumed that in any frame, the hand was the object closest to the camera, and used this assumption to segment out only the hand. Our final dataset consists of hand gestures for ten letters of the ASL alphabet bounded in a 32x32 bounding box, as shown in Figures 1 We have 1200 samples for each letter, giving us a total size of 12,000 images for our entire dataset of ten letters.

Secondly, Hand gesture images representing signs for different alphabets are taken with a Samsung DV300F 16.0 megapixel digital camera because pictures captured by a webcam are blurred. This digital camera is a CCD camera. The resolution of grabbed image is very high. Images are varied from each other in terms of format, size Fig.2(b):

and resolutions. Most of the gestures/signs are performed using both hands. Each gesture is performed at various scales, translations, and a rotation in the plane parallel to the image-plane. Since we are assuming that there is no object in the image, other than the hand gesture we need a uniform colour background for the ease of segmentation. The signer is required to wear a full sleeve black T-Shirt and a black bandage around the wrist. This provides uniform colour background for the experiment. The input hand gesture images were then transferred from the digital camera the computer for further processing.

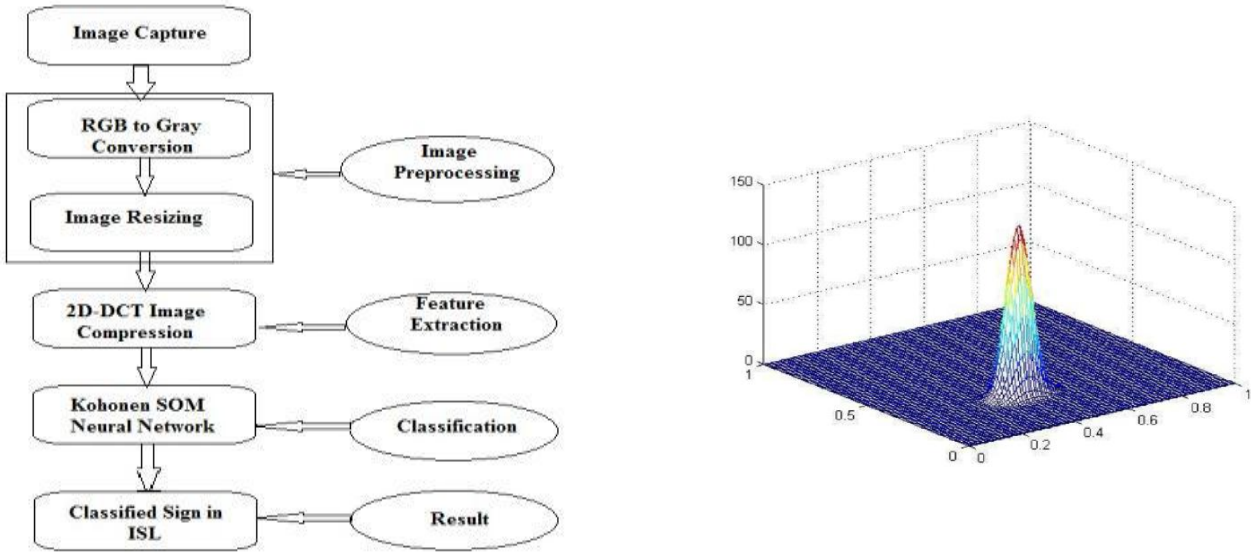


Fig.3: Schematic View of Proposed Hand Gesture Recognition System

Unsupervised Feature Learning and Classification

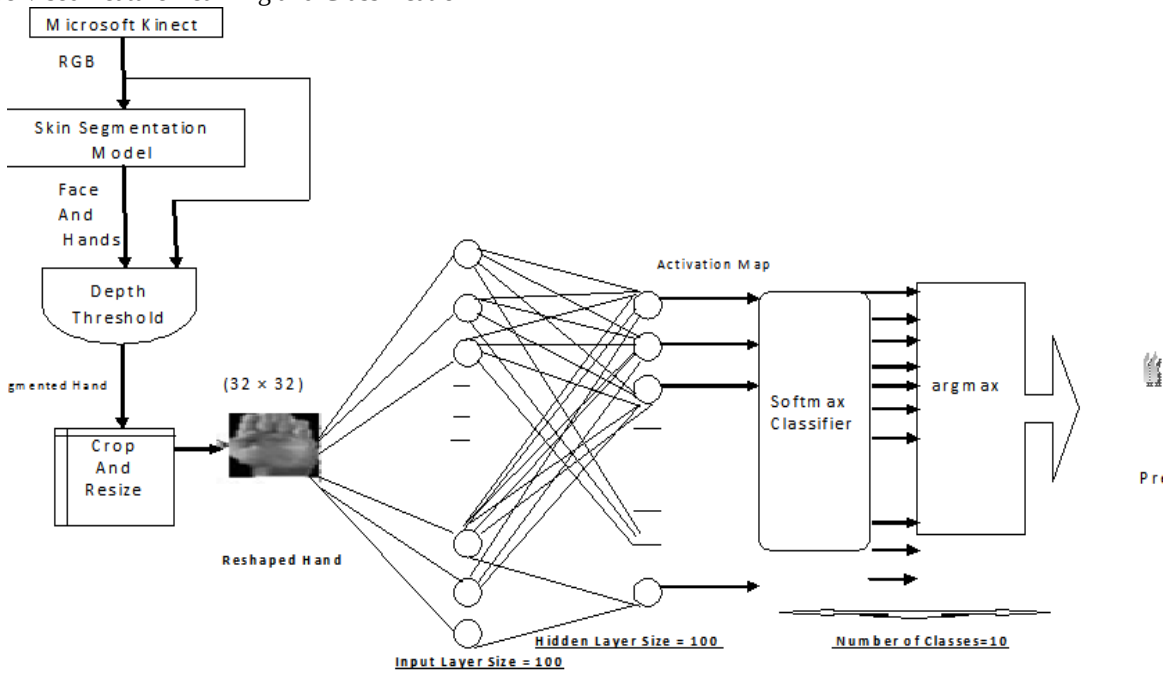


Fig.4: Block diagram for sign language recognition using unsupervised feature learning algorithm

The extracted data of hand images was fed into an autoencoder in order to learn a set of unsupervised features. We used 600 images of each letter (so, a total of 6000 images) as training data samples and fed them into the sparse autoencoder. Each input image from the segmentation blocks are images of size 32x32 pixels. A sparse autoencoder is chosen with an input layer with 32x32 nodes and one hidden layer of 100 nodes. We used L-BFGS to optimize the cost function. This was run for about 400 iterations to obtain an estimate of the weights. Visualization of the learned weights of the autoencoder is shown in Fig.5

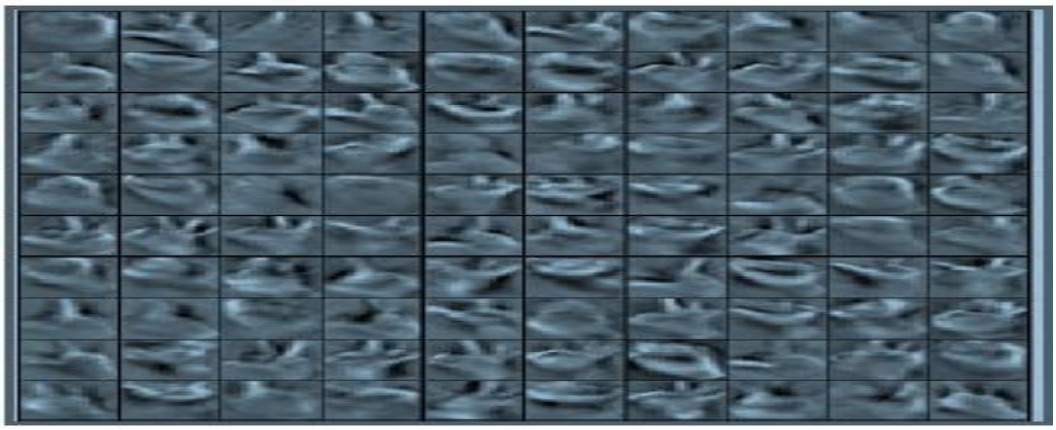


Fig.5: Visualization of the learned weights of the autoencoder

From this figure, it becomes evident that the autoencoder learns a set of features similar to edges.
The next step was to train a Softmax classifier in order to classify the 10 different letters based on the features learnt by the autoencoder. The activations of the hidden layer of the autoencoder were fed into a Softmax classifier. The Softmax classifier again learns using the L-BFGS optimization function. This algorithm converges after about 40 iterations. We tested the system classification accuracy by using the remaining 600 images per letter (so a total of 15,600 images) as our test set.

SELF ORGANISING MAP (SOM) ALGORITHM AND CLASSIFICATION

Pattern classification can be defined as the categorization of input data into identifiable classes by the extraction of significant features or attributes of the data from a background of irrelevant detail. In this paper, we have used self-organizing map (SOM) using an unsupervised learning technique in Artificial Neural Network (ANN) is used to classify DCT-based feature vectors into groups to classify whether the sign mentioned in the input image is "present" or "not present" in the ISL database. Self-Organizing Map (SOM) is one of the most popular neural network models. It belongs to the category of competitive learning networks. It is based on unsupervised learning, which means that no manual intervention is needed during the learning and that little need to be known about the characteristics of the input data. So we can use the SOM for clustering data without knowing the class memberships of the input data. The SOM can be used to detect features belonging to the problem and thus has also been called SOFM, the Self-Organizing Feature Map (SOFM). We have used the particular kind of SOM known as a Kohonen Network in this paper. This SOM has a feed-forward structure with a single computational layer arranged in rows and columns. In the architecture each neuron is fully connected to all the source nodes in the input layer –

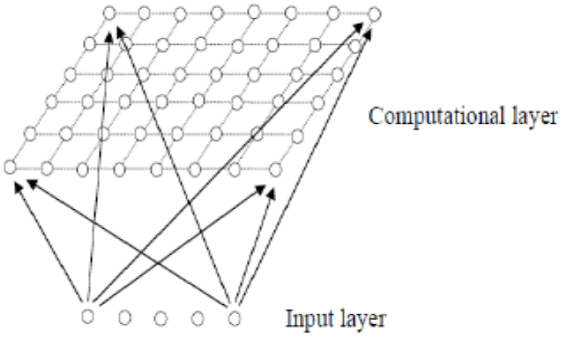


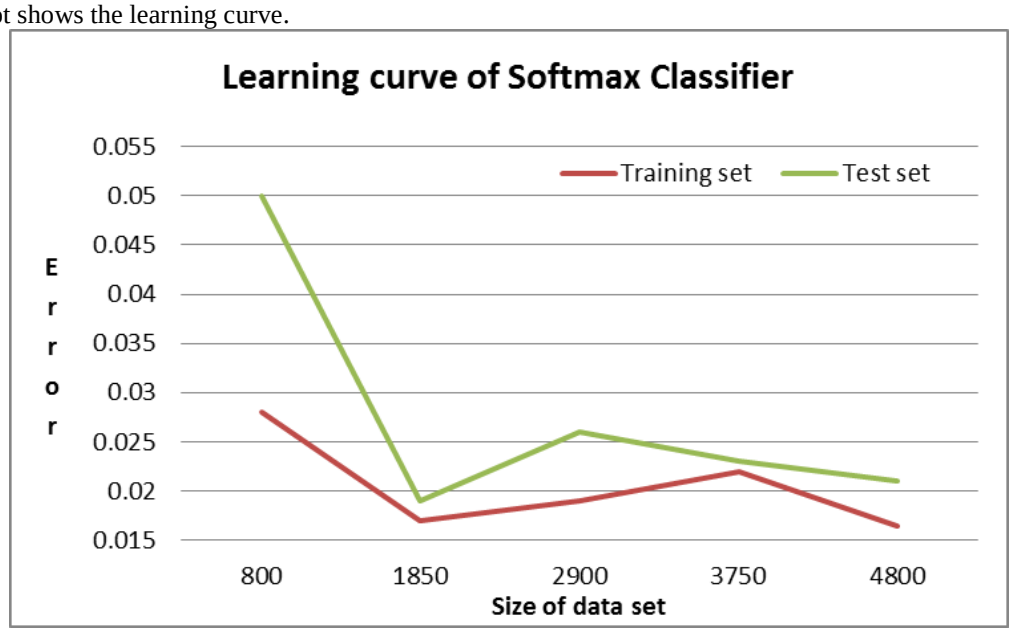
Fig.6: Self-Organizing Map (SOM)

The self-organization process involves four major components: (a) Initialization- All the connection weights are initialized with small random values.(b) Competition- In this step for each input pattern, the neurons compute their respective values of a discriminant function which provides the basis for competition. The particular neuron with the smallest value of the discriminant function is declared the winner.(c)Cooperation: In this step, the active neuron determines the spatial location of a topological neighbourhood of excited neurons, thereby providing the basis for cooperation among neighbouring neurons. (d)Adaptation- In last step, the excited neurons decrease their individual values of the discriminant function in relation to the input pattern through suitable adjustment of the associated connection weights, such that the subsequent application of a similar input pattern is enhanced.

For hand gesture recognition based on minimum Euclidean distance, as the similarity measure - training and testing for this system is performed using the MATLAB Neural Network Toolbox process, trained images are reconstructed using weight matrices and recognition is through untrained test images.

IV. RESULTS AND DISCUSSIONS

In this section, we report the performance of our system through tables and figures. Here we plotted a learning curve showing the percentage training error and test error in classification as a function of the size of the training set. The following plot shows the learning curve.



Hence suspecting that we might be in the high bias region, we decided to increase the size of our features by increasing the number of hidden units of our autoencoder to 100. Our final classification accuracy achieved using this 100 length feature vector was 98.2%.The following table summarizes all our implementation details and reports the accuracy obtained

Table I

Size of training set	Size of feature vector	Number of classes (letters)	Accuracy of classification (%)
1200	100	10	95.62
2400	100	10	98.18
3600	100	10	97.47
4800	100	10	97.92
6000	100	10	98.20

In SOM (Self organising map) algorithm Training time depend up on the number of epochs used for training. The aim of this test is to reducing training time while maintaining the accuracy rate.

Table II: Reducing processing time based on epochs

Number of Epochs	Network Training Time (in seconds)	Recognition rate (for 10 trials) approx.
50	5	60%
100	7	60%
500	32	70%
1000	72	80%
1500	123	80%
2000	179	80%

V. CONCLUSION

There are several algorithm to examine image processing and neural network as a tool for the conversion of sign language gesture into digital text .But here we want to use Unsupervised feature learning and Self organizing map (SOM) algorithm for pattern recognition of some alphabet and compare it.

Possible extensions to this project would be extending to gesture recognition to all alphabet and other non-alphabet gesture as well.

REFERENCES

[1] S.N. Deepa, B. Aruna Devi, "Second order Sequential Minimal Optimization for Brain tumor classification" , European Journal of scientific research, ISSN 1450-216X Vol. 64 No. 3 (2011)pp. 377-386.

[2] Ahmad Kharbat, Karim Gasmi, Mohamed Ben Messaond, Nacera Benamrane , Mohamed Abid , "A Hybrid Approach for Automatic Classification of Brain MRI using Genetic Algorithm and Support Vector

[3] Xiaolong Teng, Biani Wu, Weiwei Yu and Chongqing Liu. A Hand Gesture recognition system based on local linear embedding, April 2005

[4] El-Sayed Abdal Wahed, Abrahim Al Emam, Ami Badr, "Feature selection for Cancer Classification: An SVM based approach", International Journal of Computer applications (0975-8887), Volume 46- No.8, May 2012-07-23

[5] Vapnik, "The Nature of Statistical Learning Theory", Springer, New York 1995.

[6] B. Scholkopf, S. Kah-Kay, C. J. Burges, F. Girosi, P. Niyogi, T. Poggio, and V.Vapnik, "Comparing support vector machines with Gaussian kernels to radial basis function classifiers," IEEE trans Signal Processing, vol. 45, pp. 2758-2765, 1997.

