

Automated Real-time Gesture Recognition using Hand Motion Trajectory

Sweta Swami¹, Yusuf Parvez², Nathi Ram Chauhan³

^{1*2 3}Department of Mechanical and Automation Engineering,

Indira Gandhi Delhi Technical University for Women, Delhi-110006 (India)

*Corresponding Author, M.Tech (Robotics and Automation), IGDTUW, Delhi-110006 (India)

¹sweta.swami9@gmail.com

²parvez_yusuf01@yahoo.com

³namchauhan@gmail.com

Abstract—In this quite busy and technologically evolved world, gesture plays a very vital role in person's everyday life to convey the data or send command to the machines using only the motions or wave of the hand and thus, automating the processes. Gesture recognition is basically a part of HCI (Human Computer Interaction). In the past recent years, many algorithms and methodology have been implemented on the gesture recognition and achieve a touch less environment between the computer and the human. In many of the developed algorithms and the methodologies, the use of high end cameras is mandatory. For example, the Kinect camera for motion capture used in PS3 gaming. In this paper, focus has been given on the utilization of the normal web camera to recognize the gesture as robust and correct as possible. The proposed methodology in the present work is the use of deep learning algorithm in order to learn the features of the gestures and then further classify them correctly in the real time. The main target is to recognize the alphabets of the English language like (A, B, C, D) by just waving hand in front of the web camera of the laptop with 5 mega pixel resolution. Firstly, the object of any specific color is detected for tracking the movement. Then, the skin color detection is done to effectively track the movement of human hand. The Deep Belief Neural Network is used to learn the gestures in the training phase of the project. The training of the system is done using the database manually created which consists of 11 characters, 4 samples each. Finally, the gestures are recognized in real-time using Deep Learning Algorithm. The proposed methodology with incorporation of the Deep Belief Neural Network learning method achieves the 95.5 % of the success rate. The recognition rate obtained from the present work mentioned in this paper is 95.5% which is significantly higher as compared to already published recognition rate which is 92.3% on the real-time gesture recognition.

Keyword - Hand Gesture Recognition, Real time implementation, DNN, deep learning

I. INTRODUCTION

The impediment of virtual environments carries in the entire new set of complications for user interfaces. The introduction of 3D objects and world in which the occupied user permit the people like doctors, engineers, architects and scientists, to visualize complex structures and systems with prominent degrees of superiority and naturalism. A stereo or 3D view of the scene is furnished by Shutter glasses, which is no longer kept in a desktop monitor but may be a big room or projection screen. The regulating elements in such systems presently is the important communications. Virtual environments look for creating the world where the communication skills are real. Presently mechanical, magnetic and acoustic input devices track the user and deliver govern the program, selection, and running of objects in virtual scenes.

The main limitation of this technology is its cost. However, size, physical location requirements, and 2D version are other limitations. Other more advanced devices are offered for virtual reality including gloves or wearable tools like mechanical sensors, micro cameras and actuators which can handle 3D worlds, and some applications tactile sensations are also provided. Unfortunately, the user acceptance is confined due to high costs. So, making them more desirable for applications like remote surgery equipment or flight simulators.

To communicate with the factual world the new and advanced alternative, natural interface should be invented. Thus, the user must be capable to grab, point and move 3D objects just like real objects. So, a new way has been provided by these challenges for human-computer communication via computer vision techniques and it is likely to form innovative input devices which can be employed and the new devices upgraded further. Input command is given by it rather than photo or record video. Computer vision devices are transformed into input command devices as keyboard or mouse. A hand is used to create the gesture and provide an input command to the computer vision devices. The computer recognizes the signals as an input and that will be beneficial to all the users to do so without a direct device and can do what they want.

So, it makes more suitable to the computer users than keyboard or mouse. In future, vision-based interpretation devices can replace the keyboard and mouse and even exclude their usage to communicate with a computer or laptop. The communication between humans is accomplished through different media like facial, body expressions, gesture and speech. The main advantage of utilizing hand gestures to communicate or interact with a computer is the vision based non-contact input method rather than the contact based techniques.

Being an exciting part of the Human computer communications hand gesture recognition requires to be robust for factual life presentations, but human hand's complex structure presents various difficulties for being traced and deduced. Other than the gesture there are many problems like flexibility and variability of human hand structure and the shape of gestures are also included. The ability to accurately detect and recognize the gestures in spite of the variations in clarity conditions and background noise has led to the aim of evolving a robust vision based application to run the objects via hand gesture recognition. The application presents extra effective and friendly devices for human computer communications cleverly with hand gestures. Hand gestures replace the use of mouse to control the movement of the virtual object. But the complexity is involved in detection. Also, a noisy environment that creates a major impingement on the recognition performance and the detection of human hand gestures.

So, it is necessary to design low cost system using just a webcam so as to capture hand as input. Running of virtual objects can be done by modeling some predefined hand gestures based commands. The user can perform various actions as a command in a smart system which can be executed to fulfill the requirements of the user by executing them into real-life requirements.

The work in this paper is divided in three stages. 1) Feature Extraction 2) Training 3) Testing. In Section II, the work done by various researchers in the field of gesture recognition is explained. Section III describes the complete technique and methodology used for the process of real time gesture recognition. In Section IV, the results obtained from the proposed technique are presented and at the end, the conclusion of the work is shown in Section V.

II. RELATED WORK

In this section, the work done in the last few years in the field of human computer interaction via gesture recognition is examined.

Table 1: Comparison of various techniques for hand gestures

METHOD	ACCURACY	PURPOSE
HMM for data glove.	93.7%	Reduction in Spatio-temporal variability.
Multi-scale Model for gestures.	88%-90%	Segmentation and recognition of the hand.
Accelerometer-Based recognition, k-means and FFT algorithm.	Upto 91%	Recognizing, normalizing and filtering of gestures.
Novel hand gesture recognition scheme, SVM classifier.	94%	3-dimensional hand gesture recognition.
CAMSHIFT and Principal Component Analysis algorithm.	93.1%	Recognition, Segmentation and normalization of hand gestures.
Discrete Hidden Markov Model.	93.84% - 94.34%	Recognition for dynamic hand gesture.
Finger Earth Movers distance metric method.	Upto 93.2%	Recognition of only fingers of hand.

In table1 the comparison of different techniques of hand gesture recognition is noted. The various techniques with their recognition rates marked is also listed in the table.

Table 2 illustrates the various video processing techniques which can be used to process the frames of the videos which are handy in the processing of the frames taken from the camera in real time.

Table 2: Techniques for processing video.

METHOD	ACCURACY	PURPOSE
Video in painting	Average	Foreground gap filling.
Graph based region segmentation, with priority function	High	Reduction in the number of gap filling iterations.
Best first algorithm, Block based algorithm	High	Filling the gap for large objects.
Pix Mix algorithm, Pixel based approach	High	Changing live video streams and increasing the accuracy for applications in real-time.
Background subtraction method along with region centric approach.	Average	Region based gap filling.
Background and foreground method and classifiers.	Average	Reduction in false object detection.

Keskin et al. [1] proposed a system that recognize the gestures made by human hand and captures the gestures of the user with coloured gloves. 3D Kalman filters and two coloured cameras are used to remove noise for 3D reconstruction. For 3D dynamic gesture identification, Hidden Markov Model was used. HMM helped in reducing spatio-temporal variability. The difference in the shape and duration of every gesture while capturing its image is referred as the spatio temporal variability. This system divided into two parts. First is segmentation in which HMM is used the start and end points are difficult to discover. So, the model reconstruction was required with Baum-Welch algorithm. The gesture spotting implementation by Adaptive Threshold model is the second part. For 160 trails of 8 defined gestures 98.75% accuracy was achieved.

Hong et al. [2] taken into consideration the gestures affected from spatio-temporal variations. The recognition of such continuous gestures has been done after their segmentation. Two variations of gestures were studied here. First one was for outline extraction taken from two arm movements and the second one gave feature vector from a single hand movement. A Multi scale Gesture Model was also proposed. Three approaches which differ in end point localization were presented by this model. The first approach was used to find the end points by multi-scale search and motion detection strategy. The end points of fingers were located approximately with active time covering by using the second approach. Third approach is based on active programming.

The recognition ratio of 88% to 96% has been achieved for hand gesture recognition by using mentioned three approaches. But it is found that for hand gesture recognition in continuous video streams, the third approach was the best one.

Cosio et al. [3] has done the classification and gesture recognition by using Artificial Neural Network. The Wii remote was used for the recognition of gesture by rotating in X, Y, Z directions. The gesture recognition was processed in two levels so as to minimize the memory consumed and the computational cost. Accelerometer-based gesture recognition method is used and verification is done in first level. In second level, Fuzzy automata algorithm has been proposed for gesture recognition without any kind of signal processing. After this, the k-means and Fast Fourier Transform algorithm was used for filtering and normalization of the data. The recognition accuracy increased up to 95% by using Dynamic Bayesian Network.

Dominio et al. [4] introduced a novel hand gesture recognition scheme which utilizes the depth information of the image taken from the depth cameras. A set of 3-Dimensional features were used to recognize properly complex gestures by using 3-D information. Three main steps were presented in this hand gesture recognition system. In the first step, the hand samples were segmented from the background where palm, wrist and the fingers are subparts of the segmented hand samples. The four types of features are consisted in the proposed hand gesture recognition. Mentioned features for the segmentation are extracted in the second step. The distance from elevation of fingers tips to palm centre is the first two set of features. Computed curvature features of hand outline are included in the third feature set. The fourth set is based on geometry of the palm region. For the performance in front of the camera to identify the hand gesture the SVM classifier with constructed feature vectors is used and 95% accuracy is achieved.

Peng et al. [5] proposed a system by which daily information of hand movements is saved from the internet. The analysis of main components was used for hand identification by taking YCBCR colour spaces for detection of skin colour and to detect and track the hand gestures CAMSHIFT algorithm is used. The skin detection was utilized for detection of the position and region of the hand. Detection of the skin region remains continued until the tracking trigger condition is enough. The PCA was used for segmentation and

normalization. By experiments it is proved that for hand gesture recognition the accuracy rate 93.1% is achieved. The total time between 0.1sec to 0.3 sec was taken for processing a single frame.

Gharasuie et al. [6] proposed a system to identify the numbers- 0 to 9 by using active hand gesture. This involved two steps. First one is pre-processing and classification is the second step. According to Gharasuie, there are two classes of gestures- Link gestures and Key gestures. For identifying the link gestures in continuous gestures' stream, the key gestures are used. For classification, discrete HMM was used and also to find the path between any two points in continuous gesture. The Baum-Welch algorithm was used to train the DHMM. Average identification rates were obtained by using HMM (93.84% to 97.34%).

Ren et al. [7] made a robust gesture identification system which was part based as it only took into consideration the fingers of hand by using economical depth camera i.e., Kinect sensor. For Kinect sensors, it is hard to identify the hand because of its low resolution. But the large objects are captured easily by them. To deal with the captured noisy hand by Kinect sensors, here a distance metric known as Finger Earth Movers Distance was proposed to match the fingers. To manage the noise associated while detection, whole hand was not considered. Since FEMD could differentiate even the smallest differences in hand gestures, the system proved to be working efficiently in environments which are uncontrolled. The experimental results show the accuracy of 93.2%.

Different approaches and different algorithms have been studied. From the past few years, the different research done on the real-time gesture recognition is scrutinized. The main thing is that by using the high-end cameras like Kinect and the 3D depth cameras the Gesture recognition with HMM and SVM algorithm giving good results. But the use of 3D cameras and special cameras requires high investment. The main aim of this project is to recognize hand gestures with high success rate and low cost as well as the same time. So, to full fill the gap of implementing the robust recognition system of gesture with low cost camera like web camera is the main aim.

After going through the work done earlier in this area, the problem is stated for this paper work. "Hand Gesture Recognition Using Camera" is based on concept of Image processing. In recent years, a lot of work is done on gesture recognition using Kinect sensor on using HD camera but camera and Kinect sensors are costlier. The present work is focused on reducing the cost and improving the robustness of the proposed system using simple web camera. To identify the character drawn using the simple hand motion is the main target. To analyze the hand movement skin based object detection and tracking of its movement with respect to each frame is done.

The main objectives of the present work are:

1. Creating a robust Gesture recognition algorithm for real time application using web camera and MATLAB software tool.
2. Recognition of at least four characters like ('L', 'Z', 'O', 'N') will be the part.
3. Utilizing DBNN – deep learning method for the classification and learning of the motion happening in front of the camera.

Moving on to describe the methodology and computational technique in Section III.

III. METHODOLOGY

Almost all of the gesture recognition methodologies have three major steps of implementation. First step is the detection of the object which can be hand or any specific object. The main motive of this stage is to identify the object in the image to track further. Various constraints related to environmental noise are required to solve to make sure that the contour is there on the object to be detected, in our case its hand. If the hand is extracted precisely then recognition accuracy could be better. Common image problems contain variation in the illumination, poor resolution and noise. The noise less environmental condition and better resolution camera devices can improve the problem related to get better recognition. However, when the gesture recognition system is working in the real environment, it is hard to control these conditions. Hence, the image processing methodologies are a good solution for the image related constraints to develop adaptive and robust gesture recognition system. The motive of this paper is to provide real time motion as the gesture identification. There is no image specific criterion to choose the gesture. Here, the hand movement in a specific arrangement is used to detect and recognize the gesture.

The complete system steps for implementation of real-time gesture recognition are shown in the figure 1. The complete flow is also basically divided in to two parts. One is the training part and another is the testing part.

The sequence is as follows:

1. Image acquisition device is on and continuously frames are captured.
2. As soon as the motion detected with the skin region. Then the algorithm of skin detection and making masked images is executed.

3. Tracing of the hand portion is done to mark the gesture.
4. If the gesture is drawn perfectly then it is saved for the further processes
5. Next step is the feature extraction
6. After the feature extraction of the gesture the features are saved in the .mat format
7. After all gesture feature extraction and saving of the database the deep neural network is trained for those collected features for respective gesture.
8. Then in the next step, in run time a gesture is drawn and the trained neural network predict about which particular gesture it is.

The real-time hand gesture recognition system which is proposed in the paper have certain environmental condition requirement which is to be fulfilled to achieve the accurate results with better success rate.

The main problem while working on the real system is to grab the noise free skin color and extract the hand out of that skin color. Now for the better detection of the hand a color based approach is used with as less as possible background stability. That means the background should not have much of the same color as that of the skin. Our second approach is the specific color object tracing in which an object of any specific color which is not there in the background is used to trace the gesture and then identifies it. If the gesture is drawn perfectly then our approach is robust as it maps the gesture drawn on the white background image with blue color gesture drawn.

So, to get the maximum performance out of the proposed methodology, the background should be as stable as possible with less noise.

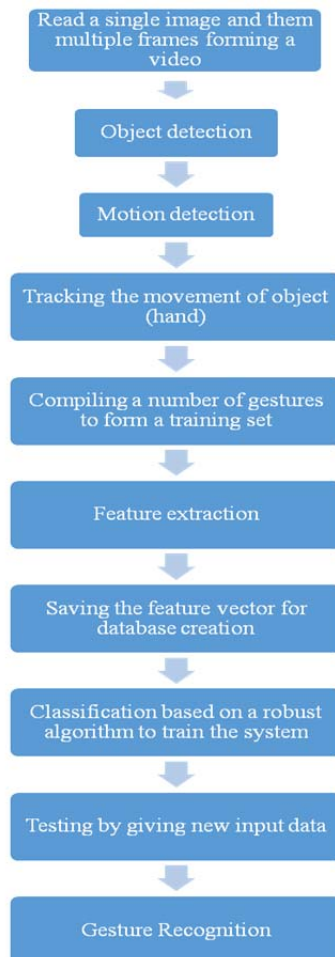


Fig 1: Steps of implementation for gesture recognition

The computational technique used in this paper work is described below.

A. Restricted Boltzmann Machines

A Boltzmann machine containing connections between a hidden and a visible node gives rise to the Restricted Boltzmann machine (RBM). A simplistic rendering of an RBM with six visible nodes and four hidden nodes are shown in fig 2.

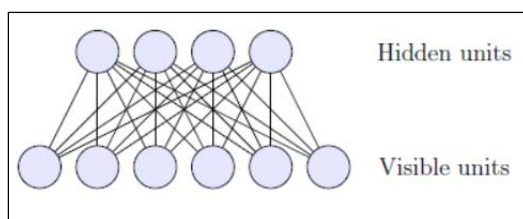


Fig 2: Restricted Boltzmann Machine [12]

RBM's can themselves be used as classification, regression, or generative models. RBMs are used by RBMs adding either a single regression label or a class SoftMax label to the visible units for supervised learning. Representative samples of the data distribution given a clamped visible label unit by a generated trained model. The RBM is mostly used as a building block of a DBN, trained in an unsupervised manner.

B. Deep Belief Networks

In fig3, a generative learning model of Deep Belief Network (DBN,) is seen target at learning the structure of the input dataset and one or more layers of features detectors. A DBN model is a combination of directed and undirected graphical models -the lower layers are directed in a "downward" and the top layer of the network is an undirected RBM, and fashion.

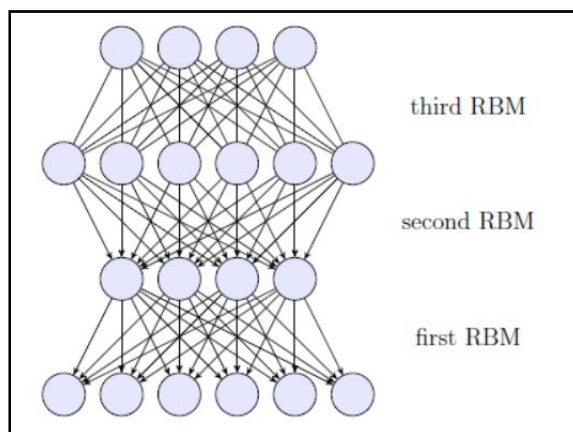


Fig 3: Deep Belief Network [8]

C. Deep Neural Networks

The paybacks of pre-training (DNN) a deep neural network has been covered broadly [9], [10], and the models discussed so far can be improved a little to construct outdated feedforward neural network. DBN trained on a specific dataset, an additional linear or logistic regression layer can be add to the top of the model, the output of the dataset is the base of training moved to the top layer of the DBN, and then use outdated back movement is designed for neural networks, and regularizing the parameter weights are included. The best results are given on the test of dataset.

D. Training a Better Neural Network

A few optimizations are there for the backpropagation algorithm covered in Nielsen [11] and used for fine-tuning the DNN. To improve learning at the top of extremes of activation functions A better error/loss function is used, and to prevent weights from growing too large weight decay is used.

E. Learning of a Network

To train a DBN is become simple by the composition of RBMs which are trained in an unsupervised way. Actually, training time of RBM governs the overall DBN training time, but simple code is made. The remarked labels are required for CDBNs during training of the top layer, therefore a training period includes that the first of all the bottom layer to be trained, for propagating the dataset via the learned RBM, and then new dataset is transformed as the training data for the next RBM. This remains active until the dataset propagated through the last trained RBM, where the labels are concatenated with the transformed dataset and used to train the top-layer associative memory.

F. Creating a Deep Neural Network

The pre-trained weights and hidden biases from a DBN, is used by the DNN model is a simple matter to extract the significant constituents from a trained DBN and the missing constituents are added to create a DNN ready to train. A DBN is taken by The DNN constructor and the number of classes in the target dataset as its arguments. The weights and biases for each layer in the new DNN are formed by these, and a final top layer with an n-output SoftMax is added, where n represent the number of classes in the target dataset.

G. Learning a Deep Neural Network

Backpropagation algorithm is used to pre-trained the top layer of weights, being initialized random weights and the backpropagation algorithm is permitted to start with close weights ideal for the entire network, and effectively random outputs are not used. But the learning algorithm for a DNN is not as space-efficient as that of the RBM, since the output of each units of each layer requires to be retained for the backpropagation algorithm. The memory usage isn't an issue with smaller batch sizes, but this is only linear growth with an increase in batch size. In its current repetition, there is no early stopping applied for the backpropagation training, but it is specified as a number of periods by the user, along with other hyper parameters as given in Table 3.

Table 3: Default DNN hyper-parameters

Hyper parameter	Effect	Default
Learning rate	Rate at which changes are applied	0.5
Batch size	Number of observations per training batch	100
Epochs	Max iterations to train over the training set	100
Lambda	L2 regularization constant	0.1

Fig. 4 shows the complete flow diagram of the training and testing process using DBNN (Deep Belief Neural Network). The input image is of size 28x28 resolution with total 784 pixel that will be processed in the DBNN hidden layers with three major layer process. After that a trained or learned vector is obtained. The testing process will be like the trained network will ask the query image and pass the image in to the trained DBNN network and return the probabilities of all the gestures and the maximum probability gesture number is the gesture of the query image.

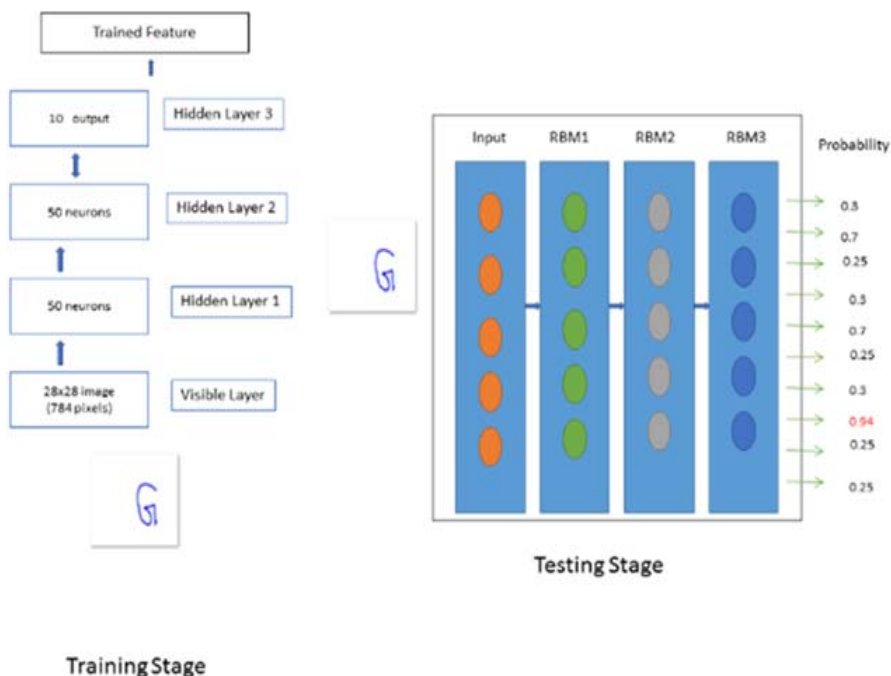


Fig4: Training and testing flowchart using DBNN.

IV. RESULTS AND DISCUSSION

A. Real time Gesture Recognition

Here in this particular work, considering deep belief neural network to recognize the gesture drawn in front of the camera in real time. To accomplish the task various preliminary experiments has been done. The various steps discussed are as follows:

1. Object detection based on color.
2. Hand detection using skin detection.
3. Training of the gestures.
 - a) Database creation
 - b) Deep Belief Neural Network training and creation
4. Testing of the gestures.

B. Object Detection based on Color

Firstly, we have done the object detection, where in we have considered an object of any specific color which is not there in the background in order to perform the color object detection algorithm. The initial screen of the algorithm is shown in the figure 5.

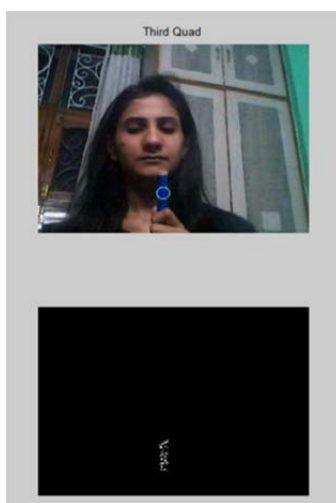


Fig.5: Object Mask image and circle draw

Figure 5 illustrates the color object detection output in which the top image is the blue color object detected with circle marked at the center image and bottom is the masked image of the same. This tracing will basically be used to mark the gesture drawn on the white background. Furthermore, the white background gesture will be treated as one of the sample to train the network to recognize the gesture.

C. Hand Detection using skin Detection

The second experimentation is done using the skin detection algorithm and detection of hand. This algorithm is used to remove the dependency of the specific color object presence.

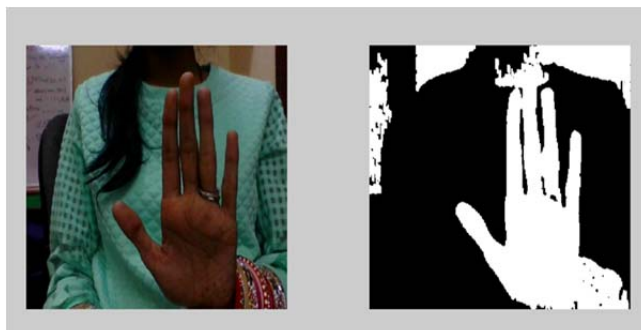


Fig. 6: Skin detection to trace the hand with noise

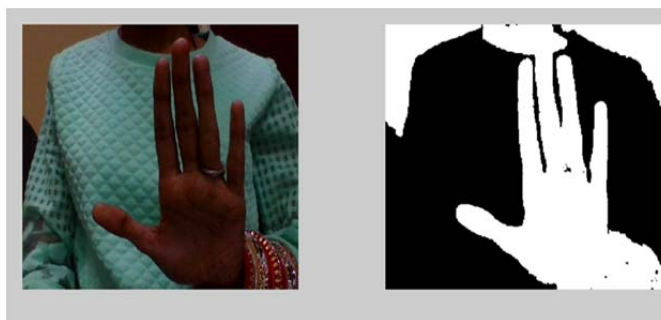


Fig.7:Skin detection with no noise

In this experiment shown in fig. 6 and 7, the skin is used as the object to be traced. And with the help of filtering methods like median filtering the and the bwareaopen function of the Matlab, the noise of the skin color is removed to get the smooth detection of the hand and later smooth tracing of the hand movement.

D. Database Creation

By using this database creation method, we have created the database of the “11” characters and saved it in their respective folders as shown in fig. 8.

s(1)	18-04-2017 11:46	File folder
s(2)	18-04-2017 11:46	File folder
s(3)	18-04-2017 11:46	File folder
s(4)	18-04-2017 11:47	File folder
s(5)	18-04-2017 11:47	File folder
s(6)	18-04-2017 11:47	File folder
s(7)	18-04-2017 11:48	File folder
s(8)	18-04-2017 11:48	File folder
s(9)	18-04-2017 11:48	File folder
s(10)	18-04-2017 11:48	File folder
s(11)	18-04-2017 11:48	File folder

Fig.8: Custom database of 11 characters is created with 4 samples of each

In fig. 9, a few samples of characters in the customized training database are shown. For each character, four samples are saved in the database.



Fig.9: Sample Database 4 samples of each character

E. Deep Belief Neural network creation and training

Fig. 10 illustrates the complete Deep learning neural network with autoencoders and the SoftMax layer.

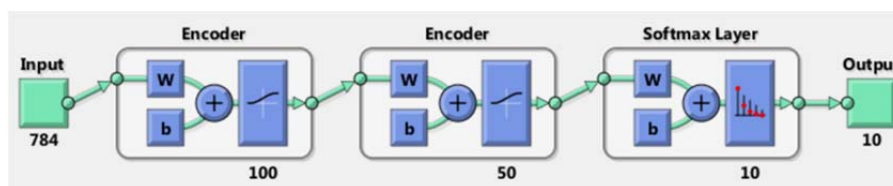


Fig. 10: Deep Learning Neural Network

Autoencoder consists of an encoder and decoder in each of its single units. The input image is of size 28x28 which gives 784-dimensional vector as the training data as shown on the input side of fig. 10. The dimension value is reduced to 100 in the first stage of neural network. This is done with the help of the default values of weights (w) and biases (b). After passing them through second hidden layer, they are again reduced from 100 to 50 dimensions. This means that the features are compressed as we move on to each successive hidden unit of the deep learning network thereby, reducing the complexity. In these hidden layers, unsupervised training is done through the encoders. The final layer, generally referred to as SoftMax layer, can be trained for the classification of the input vectors received from the previous layer into various character classes. As shown in fig. 10, 50-dimensional vectors are classified into 10 different classes which gives the desired output.

F. Testing of the Gestures

Another GUI for the testing of the gesture is created. In this GUI, as soon as we hit the start button a camera interface will get open and the time gets started. The timer we set is for 10 seconds. So, we have 10 seconds to draw any respective gesture to get it recognized.

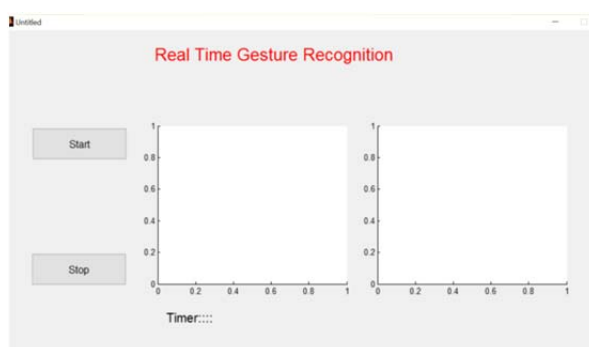


Fig. 11: Gesture recognition GUI

And for the testing we have shown the four gestures correctly recognized using the GUI, when the gesture drawn with in the 10 seconds of the time.

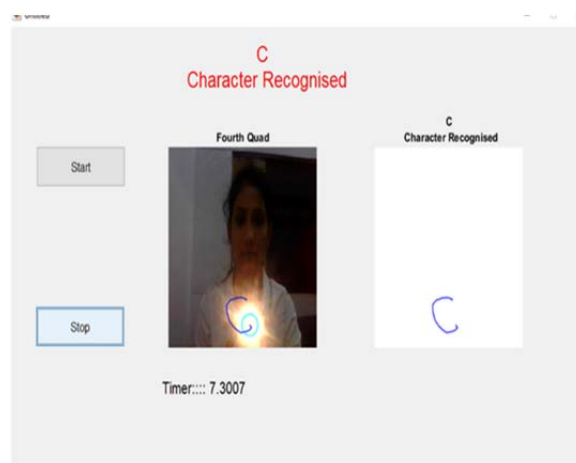


Fig. 12: Character "C" recognized. Here, an object, i.e., a flashlight has been used to test the system.

Fig.12,13 and 14 shows the testing results obtained of characters C, L and S respectively using flashlight.

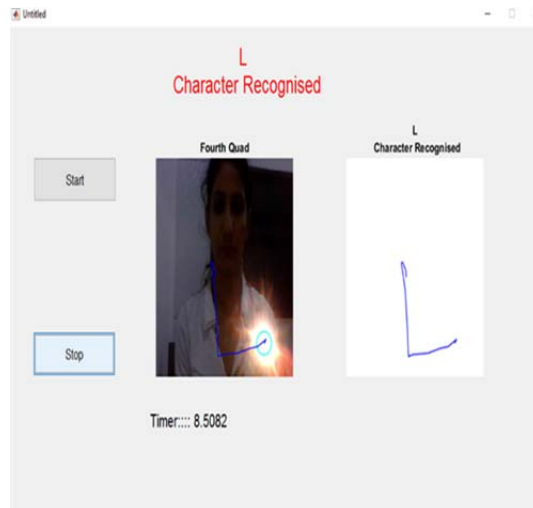


Fig. 13: Character “L” recognized using flashlight.

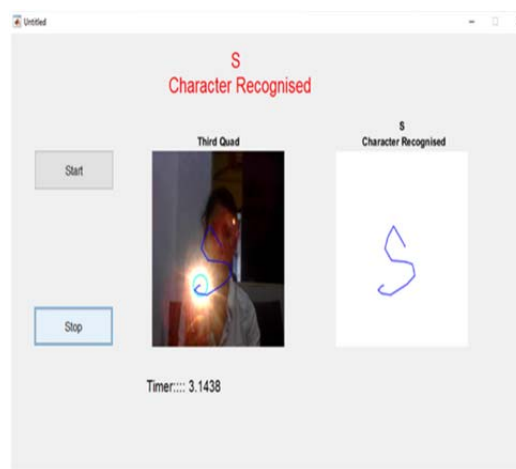


Fig. 14: Character “S” recognized using flashlight.

In fig. 15 and 16, the hand of a user is detected by the system by using skin detection method and the characters drawn are correctly recognized as shown in the figures.

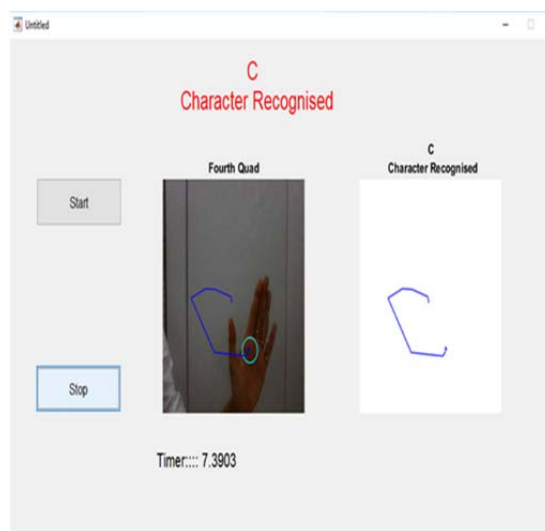


Fig. 15: Sample recognition of test image drawing character “C” using skin color detection.

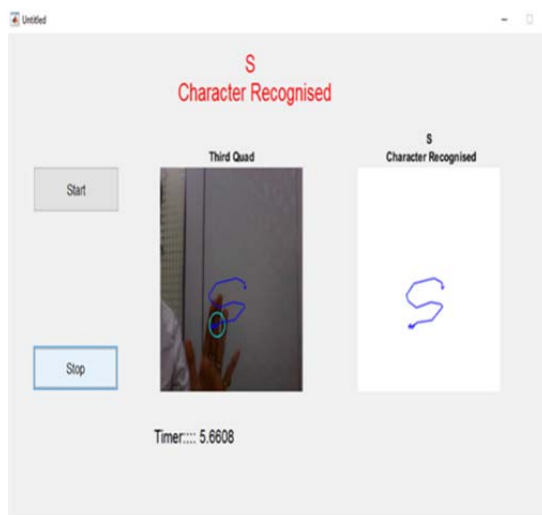


Fig. 16: Sample recognition of test image drawing character “S” using skin color detection.

Now the binary image creation is done for the hand. As shown in fig. 17,18 and 19, the skin color is detected which can be seen in the binary image where white region corresponds to the hand of the user and the gesture is recognized.

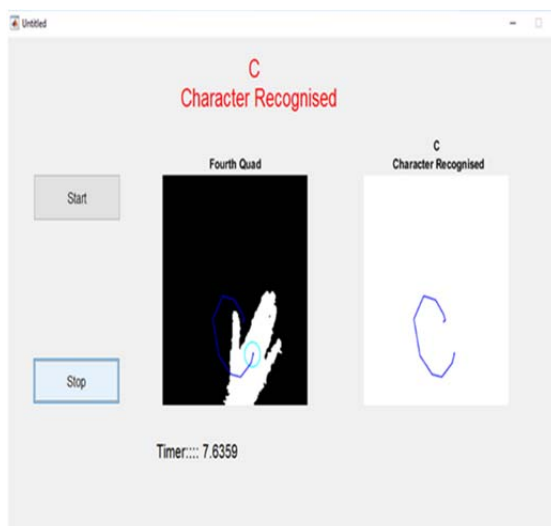


Fig. 17: Hand gesture recognition of character “C” by creating the binary image of hand.

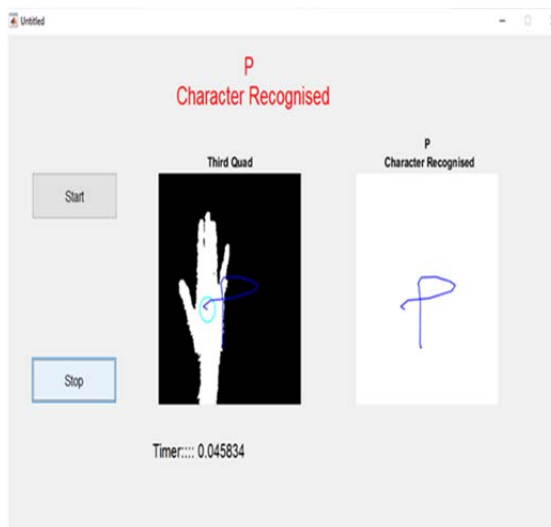


Fig. 18: Hand gesture recognition of character “P” by creating the binary image of hand.



Fig. 19: Hand gesture recognition of character “L” by creating the binary image of hand.

G. Performance Measure

The recognition of the complete system is approximately 95.5 percent. The confusion matrix shown in the figure 20 in which character wise recognition is displayed for 4 trials of each characters.

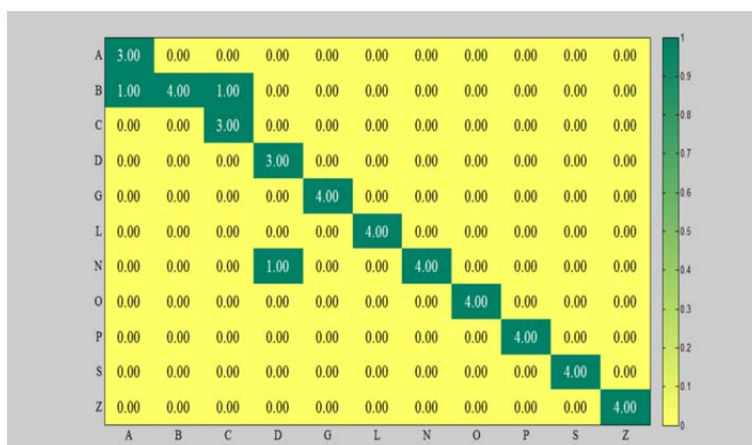


Fig. 20: Confusion Matrix

H. Comparison with published results

Table 4 shows the comparison of present work where 95.5% R.R. is achieved with the published results [13] having 92.3% R.R. for gesture recognition. This shows a major difference of approximately 3% in recognition rate validating the present work.

Table 4: Comparison of present results with published results[13]

S.NO.	METHOD	RECOGNITION RATE
1	MAHMOUD ELMEZAIN[13]	92.3%
2	PROPOSED WORK	95.5%

V. CONCLUSION

In this section, the various conclusions are listed as follows.

1. Set of 11 characters are taken into consideration in this paper and 11 out of 11 are successfully recognised in real-time implementation.
2. The task was completed using Deep Belief Neural Network technique which is quite fast and gives better results. The Deep Belief Neural Network was used to learn the gesture and then recognise the gesture in real time. The training of the system is done using the database manually created. Then the network created after the training of the system is used to detect the gesture which is quite fast and accurate.
3. The process is totally automated as the user just has to wave the hand in front of the image acquisition device. The recognition rate of our system is coming out to be 95.5 percent. The performance of the system in terms of execution time is for one character recognition the system is giving user 10 sec to wave the hand in front of the camera and draw a proper character and with in 1.12sec after the character drawn it gives the recognised character.

4. Comparing the present work recognition ratio of 95.5% with already published technique where recognition ratio was 92.3%, approximately 3% difference is achieved in the results.
5. The system can be utilised to run and control any robotic mechanism via tracking of hand gestures made by the user. Furthermore, the various processes could be automated by interfacing the system with the respective process to be controlled. For example, the recognition of character "A" can control a mechanism to proceed in a specific direction.

REFERENCES

- [1] C. Keskin, "Real Time Hand Tracking And 3d Gesture Recognition For Interactive Interfaces Using Hmm", IEEE Computer Graphics and Applications, pp. 64-71, November-December, 2002.
- [2] M.Greenspan, "Model based Segmentation and recognition of dynamic gestures in continuous video streams", IEEE International Conference on Image Processing, 2007.
- [3] B.M Lee-Cosio, "ANN for Gesture Recognition using Accelerometer Data", Published by Elsevier Ltd., 2212-0173 © 2012.
- [4] F.Dominio, "Combining multiple depth-based descriptors for hand gesture recognition" Elsevier Publications, Pattern recognitions, 2013.
- [5] S. Yu Peng, "AReal-Time Hand Gesture Recognition System for Daily Information Retrieval from Internet", IEEE Fourth International Conference on Ubi-Media Computing, 978-0-7695-4493-9/11 © 2011.
- [6] M.M.Gharasuie, "Real-time Dynamic Hand Gesture Recognition using Hidden Markov Models", 8th Iranian Conference onMachine Vision and Image Processing (MVIP), 10-12 Sept. 2013.
- [7] Z. Ren, "Robust Part-Based Hand Gesture Recognition Using Kinect Sensor", IEEE Transactions on Multimedia, Volume 15, Issue 5, Aug. 2013.
- [8] G.E. Hinton, "Training products of experts by minimizing contrastive divergence", Neural Computation, Volume 14, Issue 8, pp. 1771 – 1800, 1 Aug. 2002.
- [9] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?", The Journal of Machine Learning Research, vol. 11, pp. 625-660, 2010.
- [10] Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 5-9 March 2017.
- [11] M. A. Nielsen, "Neural networks and deeplearning", IEEE Determination Press, 2014.
- [12] A.Saran, "To recognize shapes, first learn to generate images", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 28 Sept.-2 Oct. 2015.
- [13] Mahmoud Elmezain, "Gesture Recognition for Alphabets from Hand Motion Trajectory Using Hidden Markov Models", 7th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2007.
- [14] A. Fischer, "Training Restricted Boltzmann Machines: An introduction," Neural Computation, Volume 23, Issue 3, March 2011.
- [15] Ian McLoughlin, "Robust Sound Event Classification using Deep Neural Networks", IEEE/ACM Transactions on Audio, Speech, and Language Processing, Volume 23, Issue 3, March 2015.

AUTHOR PROFILE

Sweta Swami is pursuing M.tech in Robotics and Automation from Indira Gandhi Delhi Technical University for Women, Delhi. She has done B.tech in Instrumentation and Control from Netaji Subhas Institute of Technology, Delhi. Her area of interest is human computer interaction and automation & control.

Yusuf Parvez has done M.tech from Indian Institute of Technology, Delhi. He is pursuing Ph.D from Jamia MilliaIslamia, Delhi. His research area is robotics and energy systems.

Nathi Ram Chauhan is working as an Associate Professor and Head in the department of Mechanical and Automation Engineering. He did his Ph.D. in Mechanical Engineering from IIT (Roorkee) and M.Tech with specialization in Machine Design from IIT (Roorkee). His research area is Robotics and Automation.