International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.07, March 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

Canny Edge and ANN based Object Identification and Classification

Nikhil Gala[1*], Dr. K. D. Desai[2], Arkav Banerjee[3*], Meenakshi[4], Nishi Intwala[5], Sameer Prabhu[6]

1,6 – Assistant Professor , Department of Electronics & Telecommunication,
NMIMS University's MPSTME
2 – Academic and Technical Advisor,
Guru Gobind Singh College of Engineering & Research Centre, Nashik
3,4,5 Student of B.Tech Electronics & Telecommunication, NMIMS University's MPSTME

ABSTRACT

Machine learning in one form or another is utilized in every facet of automation and industries. From social media newsfeeds, product recommendation engines, fraud detection and spam filtering, machine learning is used in abundant applications and its demand is eminent. In this paper, an object identification model was developed using an Artificial Neural Network algorithm with the aim to classify an object into its appropriate category by capturing a real time image of that object on MATLAB software. First a database was created using a white background and various image pre-processing techniques were used to make the database ready for feature extraction. After feature extraction the images were fed to the neural network toolbox as an input which was trained using supervised learning. Several real time images were tested to find the accuracy and efficiency. The results showed a 95 % accuracy for the testing images and an accuracy of 88.89% for real time images.

Index terms-Object classification, Artificial Neural Network algorithm, Pre-processing, Feature extraction, Machine Learning, canny edge detection.

I.INTRODUCTION

Machine learning is a subfield of artificial intelligence (AI), which uses past experiences and data analysis to give optimized results without human interference. It uses computational statistics to make predictions based on the database provided by the user. Machine learning uses various algorithms such as artificial neural network (ANN), support vector machines (SVM), clustering etc. Each algorithm is suitable for a specific type of application. Machine learning is divided into three main groups:

- Supervised learning: In this an algorithm is provided with example data and associated target responses that consists of numeric values or strings, so that a correct answer can be predicted when a new example is posed2 Supervised learning is very similar to the way a pupil learns under the supervision of a tutor. Good examples are provided for the pupil to memorize by the tutor, and the pupil then derives ways to solve similar problems with the help of that example.

- Unsupervised learning: In this an algorithm learns from plain examples without any associated "answers", making the algorithm decipher the data patterns on its own. It bears resemblance to the way humans make connections between objects that are similar and belong to the same class.
- Reinforcement learning: This type of learning is very much like unsupervised learning but you can give a positive or negative feedbackto the solution given by the algorithm. It is anomalous to trial and error by human beings.

Machine learning can be implemented using various software such as MATLAB, Python, Octave etc.The aim of our project is to classify simple objects into their appropriate categories using artificial neural network. This project is implemented on MATLAB software. We have used supervised learning where the "desired output" is provided by us and the computer maps the input (i.e. the object) to the correct output (i.e. the class to which it belongs to). Machine learning is used in many diverse applications especially in today's technology driven era. It is used in the biomedical domain (tumour/cancer detection) [1], security (face recognition), product recommendation which is used by various multinational companies.

II.RELATED WORK

In the paper for drowsiness detection based on single EEG channel by Ibtissem Belakhdar, Walid Kaaniche and Ridha Djmel [11] they compared the performance of two classifiers, ANN and SVM. They have used the polysonography database to obtain the feature vector, with 9 features, using FFT. These features were then fed to the ANN and SVM classifier to get a comparison between their outputs. ANN gave an accuracy of 86.5 % for drowsiness detection and 83% for alertness detection. ROC curve analysis showed that, the ANN was more robust as compared to SVM.

In a comparison of machine learning algorithms for classification of penaeid prawn species, they concluded that when compared to ANN and K-NN, SVM achieved higher accuracy for the complete set of training samples [12]. There is a small variation for smaller training sample sizes. The accuracy was found to be 94.66% for the SVM classifier, 82.66% for ANN and 79.33% for K-NN.

In a paper written by Zakareya Lasefr, Sai Shiva V N R Ayyalasomayajula, Khaled Elleithy [12],they have used an EEG data set which is available to the public after which they have eradicated the noise by putting it through three filters and then they decomposed the signal via wavelet analysis. Three features were selected for the classification procedure and One of those features is Crest Range.ANN, K-NN and SVM algorithms were then used for classifying the signals, and the highest accuracy was obtainedusing the K-NN classifier.

However, in our project we have decided to use ANN as an image classifier and found that our accuracy was higher than the accuracies shown by the artificial neural network algorithms in these papers. We are proposing that ANN classifier for object classification, as it utilizes less computational time and has a good efficiency.

III.ARTIFICIAL NEURAL NETWORK

The structure and working of the artificial neural network is similar to the animal brain, comprising of neurons that exchange information via electric pulses[7]. Like the human brain, the artificial neural network learns from previous experiences and eventually improves its performance using examples and without being explicitly programmedon the assignment of a new task [3].

An ANN contains a group of connected nodes called artificial neurons. Each connection between the nodes can transmit a signal to another node. The receiving node processes the signal(s) and then send the signal downstream to the neurons connected to it Nodes may have a state, represented by numbers. These states will always be between zero and one. They have a weight that differs every time the neural network learns something new, which increases or decreases according to the strength of the signal.

Neurons are organized in layers which perform different transformations individually as the data traverses from the input to the output layer. For better efficiency and bigger computations,Artificial Networks usually have multiple hidden layers. The rudimentary characteristics are detected by the initial layers (e.g. the pupil, the iris, eyelashes, etc...) and their computed output is fed to further layers which workout abstract generalizations (e.g. eyes, nose) until the final layers identify the complex object (e.g. face).Each link has a certain weight associated to it. ANNs have the ability to learn new tasks by altering the weights to give an optimized output. The following image shows our implementation of an ANN with 10 nodes–

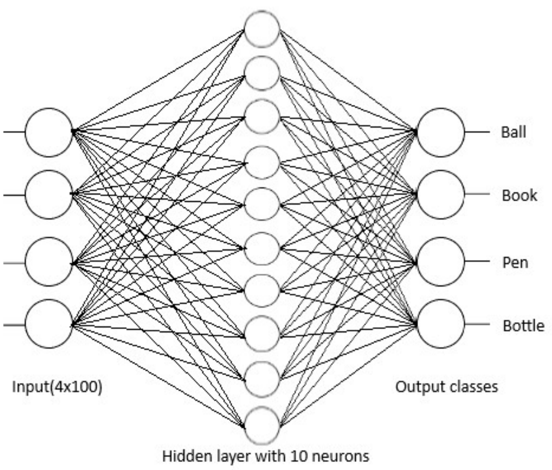


Fig. 1

There are two types of Artificial Neural Networks (ANN)– Feed Forward and Feedback.

Feed Forward ANN: The information traverses in only one direction. A node sends information to other node and does not expect any information from the other node. In this type of ANN, no feedback is provided to the input layers. These are well suited for applications which need pattern generation/recognition/classification. The inputs and outputs are fixed. Fig. 2 represents a Feed Forward ANN.

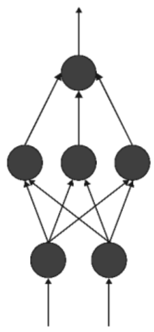


Fig. 2

Feedback ANN: In this type of ANN, the output is fed as the input to the network to attain the perfect results internally. Here, feedback loops are permitted. Fig. 3 represents a Feedback ANN.

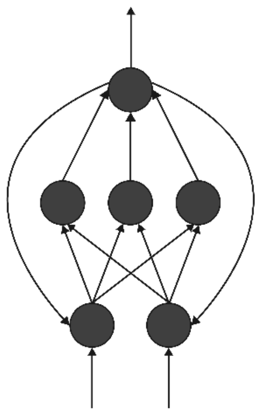


Fig. 3

In the figures above, the connection between two nodes is shown by the arrows. Each connection has a commensurate weight, which is an integer that regulates the information amongst the two nodes.If the optimized result is obtained, then there is no need to alter the value of weights. But, if an undesired output is generated, then the weights are changed in order to improve future results.The optimization of weights is achieved by backward propagation during the training phase. The weights of the links are changed by the neural network itself till the accuracy is increased to its maximum value and the predictions made are correct to a certain level. To reach a certain level of accuracy, the error in prediction is minimised across many training cycles.Training the network repeatedly will not increase the accuracy, on the contrary the network will become biased and it will fail to generalise.

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.07, March 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

The proposed object identification algorithm consists of two main stages:

(1) pre-processing and

(2) recognition of objects.

PROPOSED FRAMEWORK

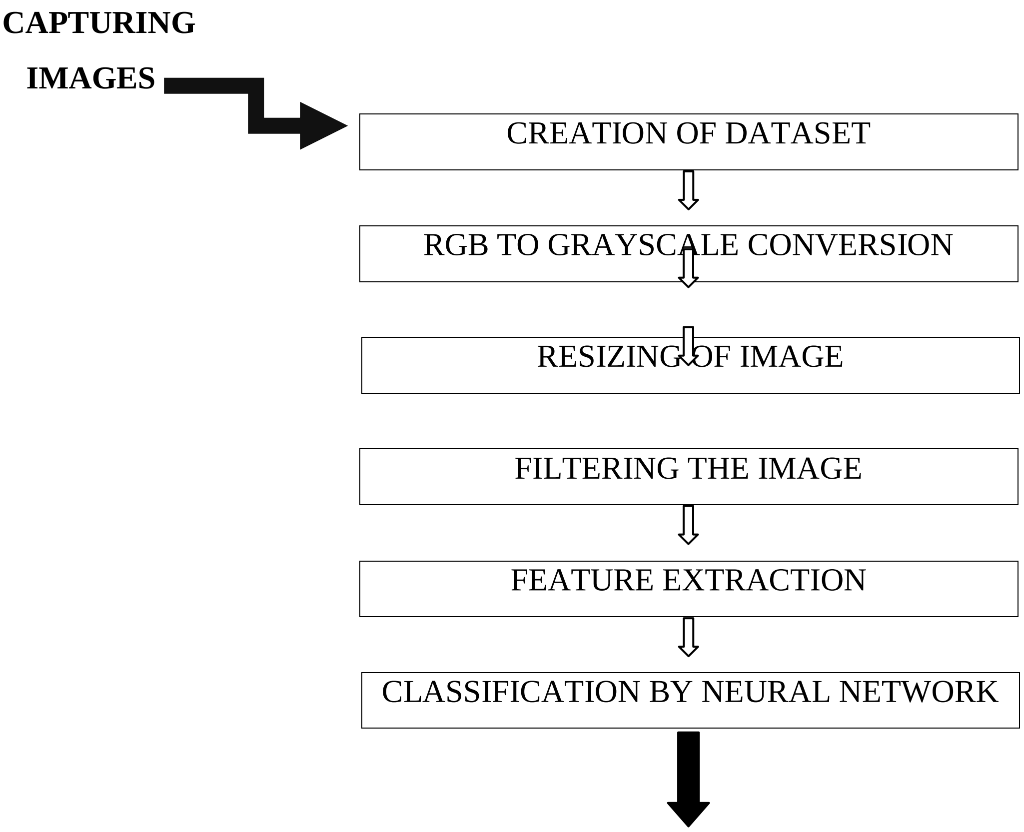


Fig. 4

IV.OUTPUT

- **Dataset creation**

Data is the main aspect of machine learning that makes the training of the algorithm possible. Data preparation is an important step in the machine learning process. This procedure consumes most of the time spent on machine learning. We have used a 9-megapixel android smartphone camera for taking pictures of the dataset. Our dataset consists of four objects, namely a ball, a book, a bottle and a spectacle. For each object we have taken 100 photos, hence a total of 400 photos. The pictures haven been taken from different angles and in different lighting conditions. In order to identify the object, eliminate any unwanted intensity changes and to get the edges of the objects, we have used a white background for every image. Our dataset consists of four hundred images in total. More the number of images, better the result will be but the computational time will increase.

- **RGB to Grayscale conversion**

Like mentioned, the objects are first captured using an android smartphone camera and stored as .jpg files. They are stored in the RGB colour format. The RGB colour model consists of the three additive colours: red, green, and blue. We have converted the RGB image into a binary image, since it is less sensitive to illumination variations compared to the RGB image. In a binary colour model, there are only 2 colours (black or white) that are present for every pixel. Hence in a lighter background, the objects have a darker foreground (or vice-versa) making it easier for detection.

- **Image Resizing**

In digital imaging and graphics, image scaling is done to resize a digital image. We can change the total number of pixels using image resizing. We have used an interpolation algorithm through MATLAB software to rescale our image from 4032 × 2268 pixels to 125 × 75 pixels. This is done in order to reduce the computational time as well as to have a uniform dataset to be used for training.

- **Image Filtration**

We have used an averaging filter through MATLAB software for removing any graininess or distortions from the background. It smoothens the image and removes noise from the background so that it is easier to detect the edges of the object only. After this the image is ready for feature extraction.

- **Feature Extraction**

Selecting good features is a critical phase in any object recognition system.In our case, we have resized the segmented binary image to 125×75 pixels. The areas where the light intensity differs and frequency changes can be seen, the algorithm is able to detect a set of connected curves called edges while preserving the important structural properties of an image [9]. This helps in filtering out the redundant data in the image. In our project we have used canny edge detector as opposed to Sobel and Prewitt as it gave us a better output for edge detection. Canny edge detection is a multistage algorithm which calculates the local maxima of the slope of $(C_{x,z})$ in order to find the edges of the image. Using the calculated slope found by the algorithm, the strong and weak edges are detected. It only includes those weak edges which are connected to strong edges. We have used Canny Edge Detection through MATLAB software in order to find the edges of the objects and to feed it to the artificial neural network.

To smooth the image, a Gaussian filter is applied initially. This reduces the noise to a great extent. The equation for a Gaussian filter kernel of size (2y+1)×(2y+1) is given by:

$$C_{x,z} = \frac{1}{2\pi\sigma^2}\exp\left\{\frac{(x-(y+1))^2+(z-(y+1))^2}{2\sigma^2}\right\};\ 1\leq x,z \leq (2y+1)$$

Whereσ symbolises the standard deviation.

Any direction may be indicted by an edge and hence the canny algorithm makes the use of four filters to detect an edge in the blurred image. For the first derivative in the horizontal direction (GIz) and the vertical direction (GIY) the algorithm can be used to return a value.

$$GI = \sqrt{GI_Z{}^2 + GI_Y{}^2}$$

$$\emptyset = atan2\lfloor GI_Y, GI_z \rfloor$$

Here, GI indicates the edge gradient while ∅ denotes the direction. The value of GI is evaluated by the hypot and the arctangent function (having 2 arguments). A specific angle value will be indicated when an edge direction falls in any of the colour regions.

V.RESULTS

We have used the MATLAB 2013b version for our project. We have created a dataset of 400 images (100 images per class) each resized to 125×75 pixels. 10 neurons have been created in the hidden layer for getting optimum results. The neural network toolbox launches a window with options for pattern recognition, neural network fitting, time series tools and clustering.

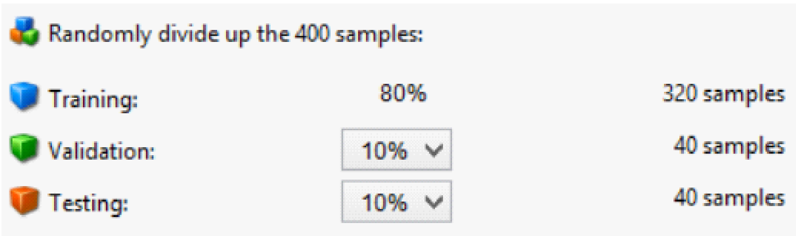


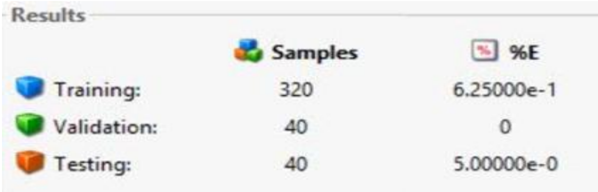
Fig. 5

Training Set: This data adjusts the weights on the neural network. Our training set contains 320 images. We can change this number according to the dataset. For an optimized result we have used 80% of the total images

Validation Set: Validation dataset is a sample of dataset images which are not shown to the model during training and is used to give an estimated accuracy of the model as well as change the hyper-parameters that is the weights to get better accuracy until the optimal accuracy is achieved. If your model is trained on a training set only, it is very likely to get 100% accuracy, thus it gets a very poor performance on test set. Thus a validation set, which is independent from the training set, is used for validating whether the model can test the dataset well.

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.07, March 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

Testing Set: 40 images which are not used in the training set are used to test the accuracy and predictive power of the system. After setting the percentages of each set, the percentage errors are computed. This shows that there was a 6.25% error in training process and a 5.00% error in testing.

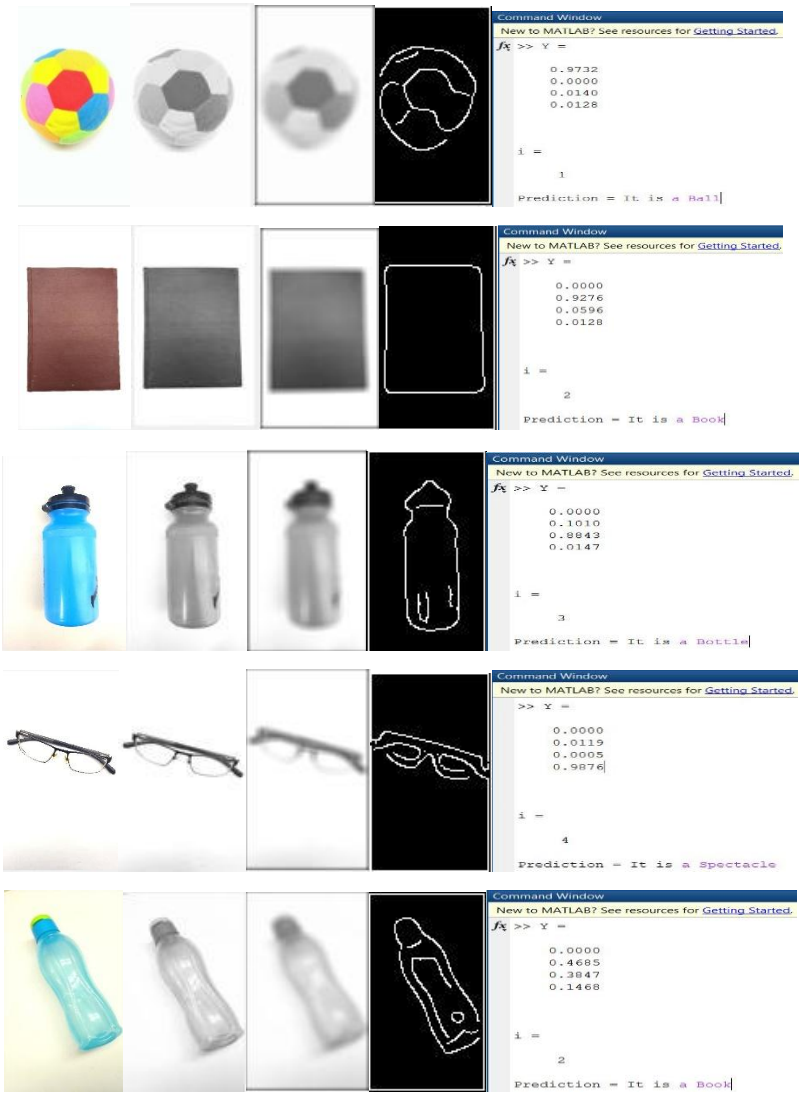


Results

	Samples	%E
Training:	320	6.25000e-1
Validation:	40	0
Testing:	40	5.00000e-0

Fig. 6

The real time images are as follows-



(A) (B) (C) (D) (E)

Fig. 7

International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.07, March 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

(A) Original image (B) Grayscale image (C) Filtered image (D) Edge detected image (E) Output window on MATLAB

After obtaining a good accuracy of 95% in the testing phase we further applied the algorithm on 18 real time images and observed that 16 of them have been classified correctly. Hence an accuracy of 88.89% has been achieved. Fig 7 shows response for 5 real time images out of the 18 that were tested. 5th image sample has been included to show where the algorithm developed could not identify the object correctly.

The results are tabulated using the confusion matrix which uses only the testing images. Referring to Fig. 6, the rows hold the values for the predicted class (Output Class), and the columns hold the value of the true class (Target Class). The oblique cells show the true positive outputs (i.e. true and predicted class are equal). The incorrect predictions are shown by the rest of the cells. The column on the right represents the predicted class accuracy, whereas the bottom row shows the precision of each true class. The overall accuracy is shown by the bottom right of the plot.

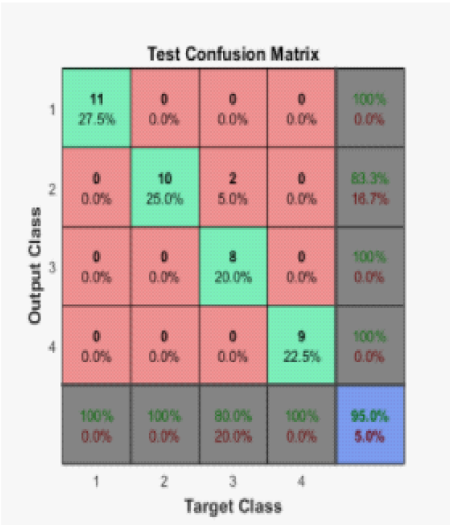


Fig. 8

VI.CONCLUSION

In this paper, 400 images have been trained and tested using an artificial neural network which was an effective algorithm to arrange them into 4 classes. Further, the pre-processing techniques such as conversion to greyscale,resizing and filtration along withcanny edge detection was an optimal decision since it gave us good experimental results. Finally, an accuracy of 95% was obtained for these objects on the testing images. When real time images were then fed into the ANN, it could classify the images into one of the four categories based on the training it had received. The accuracy for real time images was 88.89%

VII.FUTURE SCOPE

In the future, we can use convolutional neural networks and unsupervised learning for better results. We can create more classes and we can use more number of images in the dataset. Better feature extraction and pre-processing techniques such as image segmentation can be used in order to get a very high accuracy for face, characteras well as handwriting analysis.

REFERENCES

[1] Smita.A.Nagtode,Bhakti B. Potdukhe and Pradnya Morey, "Two dimensional Discrete Wavelet Transform and Probabilistic Neural Network used for Brain Tumor Detection and Classification" 2016, Fifth International Conference on Eco-friendly Computing and Communication Systems (ICECCS-2016)

[2] Md. Abdullah-al-mamun, Aminul Islam, "Emblematical image based pattern recognition paradigm using Multi-Layer Perceptron Neural Network" 2015, IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 17, Issue 6, Ver. II (Nov – Dec. 2015), PP 59-66

[3] Dr. Rama Kishore, Taranjit Kaur, "Backpropagation Algorithm: An Artificial Neural Network Approach for Pattern Recognition" 2012,International Journal of Scientific & Engineering Research, Volume 3, Issue 6, June-2012 ISSN 2229-5518

[4] L.S.Alandkar, S.R.Gengaje, "STUDY OF OBJECT DETECTION IMPLEMENTATION USINGMATLAB" 2016,IJRET: International Journal of Research in Engineering and TechnologyeISSN: 2319-1163 | pISSN: 2321-7308, Volume: 05 Issue: 08 | Aug-2016

[5] B. Chitradevi,P.Srimathi, "An Overview on Image Processing Techniques" 2014, International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 2, Issue 11, November 2014

[6] M. A. Rashidan, Y. M. Mustafah, S. B. A. Hamid, N. A. Zainuddin and N. N. A. Aziz, "Detection of Different Class Moving Objects in Public Surveillance using Artificial Neural Network(ANN)" 2014, 2014 IEEE DOI 10.1109/ICCCE.2014.75, 5th International Conference on Computer & Communication Engineering

[7] Richard P. Lippmann, "An Introduction' to Computing with Neural Nets" 1987,APRIL 1987 IEEE ASSP MAGAZINE

[8] Y.Ramadevi, T.Sridevi, B.Poornima, B.Kalyani, "SEGMENTATION AND OBJECT RECOGNITION USING EDGE DETECTION TECHNIQUES" 2010, International Journal of Computer Science & Information Technology (IJCSIT), Vol 2, No 6, December 2010

[9] Pinaki Pratim Acharjya, Ritaban Das & Dibyendu Ghoshal, "Study and Comparison of Different Edge Detectors for Image Segmentation" 2012,Global Journal of Computer Science and Technology Graphics & Vision Volume 12 Issue 13 Version 1.0 Year 2012

[10] Ranjeet Kumar Nigam, N.P. Waghmare, A.K. Gupta, "Evaluation and Analysis of Different Type of Edge Detection Techniques on Cartridge Case Image" 2014,International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169 Volume:2 Issue:10

[11] Ibtissem Belakhdar, Walid Kaaniche, Ridha Djmel and Bouraoui Ouni, "A Comparison Between ANN and SVM Classifier for Drowsiness Detection Based on Single EEG Channel" 2016, 2nd International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2016 March 21-24, 2016, Monastir, Tunisia

[12] V. Sucharita, S.Jyothi, P.Venkateswara Rao, "Comparision of Machine Learning Algorithms for Classification of Penaeid Prawn Species" 2016, 2016 International Conference on Computing for Sustainable Global Development(INDIAcom)

[13] Zakareya Lasefr, Sai Shiva VNR Ayyalasomayajula, and Khaled Elleithy, "An Efficient Automated Technique for Epilepsy Seizure Detection Using EEG signals" 2017