

# Paper Currency Recognition Using Backpropagation Neural Networks

P.P.S.Subhashini<sup>1</sup>, Dr.M.Satya Sairam<sup>2</sup>, Dr. D.Srinivasa Rao<sup>3</sup>Associate Professor, Department of ECE, RVR & JC College of Engineering, Guntur, India<sup>1</sup>Professor & HOD, Department of ECE, Chalapathi Institute of Engineering and Technology, Guntur, India<sup>2</sup>Professor, Department of ECE, JNTUH, Hyderabad, India<sup>3</sup>

**ABSTRACT:** Main objective of this paper is the currency recognition using ANN (Artificial Neural Network) with features of paper currency sample extracted using Canny Edge Histogram and GLCM. Canny edge detection histogram detects even weak edges and it has low error rate. A 1x5 edge orientation histogram is computed. As Sobel mask is applied to find the max Sobel gradient. GLCM (Gray Level Co-Occurrence Matrix) is used for analyzing the texture. It is a statistical method that considers the spatial relationship of the pixels and characterizes the texture of the image. The extracted features are used in feed forward back propagation network.

The project is implemented in MATLAB environment. Neural network uses 75% of extracted features for training and remaining 25% features for testing. The network performance is evaluated with the percentage of correct classification.

**KEYWORDS:** Artificial Neural Networks (ANN), Gray Level Co-occurrence Matrix GLCM, Canny edge detection

## I. INTRODUCTION

Monetary transactions are an integral part of our day to day activities. Automated paper currency recognition with good accuracy and high processing speed has great importance. The technology of currency recognition aims to search and extract visible as well as hidden marks on paper currency for efficient classification. The currency recognition system consists of following modules: Feature extraction, training and testing. Feature extraction deals with extraction of potential features in image based on color, texture and shape. Implementation of training and testing phase is by using ANN (Artificial Neural Network). An artificial neural network usually called as neural network (NN), is a mathematical model or computational model that is inspired by the structure and functional aspects of biological neural network. A neural network consists of inter connected group of artificial neurons.

In the proposed system a high resolution scanner is used to acquire the image. The acquired image of a paper currency is first converted to gray scaled image. Conversion to gray scale facilitates further pre-processing. The task of pre-processing is achieved by converting colored currency images into grayscale, then black-white images. After that, the edge of the image is filtered using Prewitt method. Then, the image edge is detected using Canny's edge detection method. Different currency images. This paper is meant for Paper Currency Recognition System. It is designed to just recognize four different currencies. This means that this system will not be concerned with other currencies. In addition, the system will focus on paper currencies, not coins.

Moreover, the system is not concerned with verification of the validity of the paper currencies (i.e. verifying that the paper currency is genuine and not faked). This is usually done using other methods which might involve sensing the magnetic string embedded inside the currency, or some other methods. The goal of the paper is to achieve the best accuracy in recognizing patterns with the lowest possible. Given the fact that paper currencies are usually recognized by machines that have small power (such as auto-seller machines and ATM's), the cost is a limiting factor. Therefore, it

## International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

is really urgent for all paper currency recognizers to minimize the power consumption, and, at the same time, achieve high level of accuracy.

### II. PRE PROCESSING

**A. Formulation:** Mathematically, the operator uses two 3×3 kernels which are defined in Equation (1.1) are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. If we define A as the source image, and  $G_x$  and  $G_y$  are two images which at each point contain the horizontal and vertical derivative approximations, the latter are computed as shown in Equation (1).

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} * A \quad G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{bmatrix} * A \quad (1)$$

Where \* here denotes the 1-dimensional convolution operation.

The x-coordinate is defined here as increasing in the "right"-direction, and the y-coordinate is defined as increasing in the "down"-direction. At each point in the image, the resulting gradient approximations can be combined to give the gradient magnitude, using Equation (2).

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

Using Equation (1.2), we can also calculate the gradient's direction using Equation (3)

$$\theta = \text{atan2}(G_x, G_y) \quad (3)$$

Where, for example,  $\theta$  is 0 in Equation (3) for a vertical edge which is darker on the right side.

**B. Canny Edge Detection Method:** The Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. It was developed by John F. Canny in 1986. Canny also produced a computational theory of edge detection explaining why the technique works.

Canny edge detection is a technique to extract useful structural information from different vision objects and dramatically reduce the amount of data to be processed. It has been widely applied in various computer vision systems. Canny has found that the requirements for the application of edge detection on diverse vision systems are relatively similar. Thus, an edge detection solution to address these requirements can be implemented in a wide range of situations. The general criterion for edge detection includes:

1. Detection of edge with low error rate, which means that the detection should accurately catch as many edges shown in the image as possible
2. The edge point detected from the operator should accurately localize on the center of the edge.
3. A given edge in the image should only be marked once, and where possible, image noise should not create false edges.

To satisfy these requirements Canny used the calculus of variations a technique which finds the function which optimizes a given functional The optimal function in Canny's detector is described by the sum of four exponential terms, but it can be approximated by the first derivative of a Gaussian.

Among the edge detection methods developed so far, canny edge detection algorithm is one of the most strictly defined methods that provide good and reliable detection. Owing to its optimality to meet with the three criteria for edge detection and the simplicity of process for implementation, it became one of the most popular algorithms for edge detection.

The Process of Canny edge detection algorithm can be broken down to 4 different steps:

1. Apply Gaussian filter to smooth the image in order to remove the noise
2. Find the intensity gradients of the image
3. Apply non-maximum suppression to get rid of spurious response to edge detection
4. Apply double threshold to determine potential edges

Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

## International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

Every step will be described in details as following. The introduction of procedure below is developed based on Prof Thomas Moeslund's lecture note for digital image processing in Indian Institute of Technology.

**C. Median Filter:** In signal processing, it is often desirable to be able to perform some kind of noise reduction on an image or signal. The median filter is a nonlinear digital filtering technique, often used to remove noise. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise.

The main idea of the median filter is to run through the signal entry by entry, replacing each entry with the median of neighbouring entries. The pattern of neighbours is called the "window", which slides, entry by entry, over the entire signal. For 1D signals, the most obvious window is just the first few preceding and following entries, whereas for 2D (or higher-dimensional) signals such as images, more complex window patterns are possible (such as "box" or "cross" patterns). Note that if the window has an odd number of entries, then the median is simple to define: it is just the middle value after all the entries in the window are sorted numerically. For an even number of entries, there is more than one possible median, see median for more details.

To demonstrate, using a window size of three with one entry immediately preceding and following each entry, a median filter will be applied to the following simple 1D signal:

$$x = [2 \ 80 \ 6 \ 3]$$

So, the median filtered output signal  $y$  will be:

$$y[1] = \text{Median}[2 \ 2 \ 80] = 2$$

$$y[2] = \text{Median}[2 \ 80 \ 6] = \text{Median}[2 \ 6 \ 80] = 6$$

$$y[3] = \text{Median}[80 \ 6 \ 3] = \text{Median}[3 \ 6 \ 80] = 6$$

$$y[4] = \text{Median}[6 \ 3 \ 3] = \text{Median}[3 \ 3 \ 6] = 3$$

$$\text{i.e., } y = [2 \ 6 \ 6 \ 3].$$

Note that, in the example above, because there is no entry preceding the first value, the first value is repeated, as with the last value, to obtain enough entries to fill the window. This is one way of handling missing window entries at the boundaries of the signal, but there are other schemes that have different properties that might be preferred in particular circumstances:

- 1) Avoid processing the boundaries, with or without cropping the signal or image boundary afterwards
- 2) Fetching entries from other places in the signal. With images for example, entries from the far horizontal or vertical boundary might be selected
- 3) Shrinking the window near the boundaries, so that every window is full.

Typically, by far the majority of the computational effort and time is spent on calculating the median of each window. Because the filter must process every entry in the signal, for large signals such as images, the efficiency of this median calculation is a critical factor in determining how fast the algorithm can run. The naïve implementation described above sorts every entry in the window to find the median; however, since only the middle value in a list of numbers is required, selective algorithm can be much more efficient. Furthermore, some types of signals (very often the case for images) use whole number representations: in these cases, histogram medians can be far more efficient because it is simple to update the histogram from window to window, and finding the median of a histogram is not particularly onerous.

Median filter is used for de noising the image after resizing the image. A median filter considers each pixel at the same time and looks at its neighboring pixels to decide whether it is representative of its surroundings or not. The pixel value is replaced with median of neighboring pixel value instead of replacing with mean. Median filter is more effective when the goal is to simultaneously reduce noise and at the same time to preserve edges in the image. From the input color image each Red, Blue and Green channels are denoised separately by using median filter.

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

## III. DESIGN APPROACH

Design of our project includes pre-processing of scanned images, extracting features, training and testing the neural network

**A. Image Acquisition:** In this step we scan the images of six different denominations of good quality and bad quality for training as well as for testing purpose. Fig.1:represents the scanned images of currency notes.



Fig. 1: Currency notes

**B. Pre-Processing:** Median filter is used for de noising the image after resizing the image. A median filter considers each pixel at the same time and looks at its neighboring pixels to decide whether it is representative of its surroundings or not. The pixel value is replaced with the median of neighboring pixel values. Median filter is more effective when the goal is to simultaneously reduce the noise and at the same time to preserve edges in the image. Fig. 2 shows an example of resized and median filtered images.

## International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017



**Fig. 2 Median Filtered Image**

**C. Feature Extraction:** It deals with extracting attributes that result in some quantitative information of interest for differentiating one denomination from another. A total of eighteen features are extracted from each note. The features of the image based on color, texture and shape are extracted for efficient classification. The input vector is represented in Equation (4).

$$F = (C_m, C_v, C_s, e_h, g_{lcm}) \quad (4)$$

**i. Color Features** Color is a widely used important feature for image representation. This is very important as it is invariant with respect to scaling, translation and rotation of an image. Color space, color quantification and similarity measurement are the key components of color feature extraction.

**ii. Color Moments** Color moments are measure that can be used to differentiate images based on their features of color. Once calculated, these moments provide measurements for color similarity between images. These values of similarity can then be compared to the values of images indexed in a database for tasks like image retrieval.

The basis of color moments lays in the assumption that the distribution of color in an image can be interpreted as a probability distribution. Probability distributions are characterized by a number of unique moments (e.g. Normal distributions are differentiated by their mean and variance). It therefore follows that if the color in an image follows a certain probability distribution, the moments of that distribution can then be used as features to identify that image based on color.

Moment 1 - MEAN

Mean value is derived by using Equation (5).

$$E_i = \sum_{j=1}^N p_{ij} \quad (5)$$

Moment 2 - STANDARD DEVIATION

Standard deviation is derived by using Equation (6).

$$V_i = \sqrt{\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^2} \quad (6)$$

MOMENT 3 – SKEWNESS

Skewness is derived by using Equation (7)

$$S_i = \sqrt{\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^3} \quad (7)$$

**iii. Edge Features (Canny Edge Detection):** Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image.

Canny edge detection algorithm is also known as the optimal edge detector. Canny's intentions were to enhance the many edge detectors in the image.

- The first criterion should have low error rate and filter out unwanted information while the useful information preserve.
- The second criterion is to keep the lower variation as possible between the original image and the processed image.
- Third criterion removes multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and

## International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

suppresses any pixel that is not at the maximum using non-maximum suppression. The gradient array is now further reduced by hysteresis to remove streaking and thinning the edges.

**C. Convolution:** First step to canny edge detection requires one method of filter out any noise and still preserve the useful image. Convolution is a simple mathematic method to many common image-processing operators. An example of image and kernel is shown in Fig.3

**Convolution Operation:** Convolution is performed by sliding the kernel or mask over a grey-level image. The kernel starts from the top left corner and moves through entire image within image boundaries. Each kernel position corresponds to a single output pixel. Each pixel is multiplied with the kernel cell value and added together.

Example for pixel value at position 57 is shown in Equation (8).

$$O_{57} = I_{57}K_{11} + I_{58}K_{12} + I_{59}K_{13} + I_{67}K_{21} + I_{68}K_{22} + I_{67}K_{21} \quad (8)$$

Mathematical representation of Equation (6.5) is shown in Equation (9).

$$O(i, j) = \sum_{k=1}^n \sum_{l=1}^n I(i+k-1, j+l-1)K(k, l) \quad (9)$$

The output image will have  $M-m+1$  rows and  $N-n+1$  columns,  $M$  image rows and  $N$  image columns,  $m$  kernel rows and  $n$  kernel columns. The output image will be smaller when compared to the original image. This is due to the bottom and right edge pixels, which can't be completely mapped by the kernel therefore  $m-1$  right hand pixels and  $n-1$  bottom pixels can't be used.

**E. Texture Features:** Texture is a feature that represents the surface and structure of an image or it can be defined as a regular repetition of an element or pattern on a surface. Texture of an image is complex visual patterns that are composed of entities or regions with sub-patterns with the characteristics of brightness, color, shape, size, etc.

**Gray Level Co-occurrence Matrix:** A statistical method of examining texture that considers the spatial relationship of pixels is the gray level co-occurrence matrix (GLCM), also known as the gray level spatial dependence matrix. The GLCM functions characterize the texture of an image by calculating how often pairs of pixels with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix. Table.1 lists the GLCM statistics.

**Table. 1 GLCM statistical measures**

Statistic	Description
Contrast	Measures the local variations in the GLCM
Correlation	Measures the joint probability occurrence of the specified pixel pairs
Energy	Provides the sum of squared elements in the GLCM. Also known as uniformity or the angular second moment.
Homogeneity	Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.

The GLCM which is a square matrix can reveal properties about the spatial distribution of the gray levels in the texture image. It shows how often a pixel value known as reference pixel with the intensity value  $i$  occur in a specific relationship to a pixel value known as neighbour pixel with the intensity value  $j$ . so, each element  $(i, j)$  of the matrix is the number of occurrences of the pair of pixel with value  $i$  and pixel with value  $j$  which are at a distance 'd' relative to each other. The spatial relationship between two neighbouring pixels can be specified in many ways with different offsets and angles, the default one being between a pixel and its immediate neighbour to its right.

Each element of GLCM is the number of times that two pixels with gray tone  $i$  and  $j$  are neighbourhood in distance  $d$  and direction  $\theta$ . For 00 co-occurrence matrix, there are 2 occurrences of the pixel intensity value 1 and pixel intensity value 3 adjacent to each other in the input image. Also, the occurrence of pixel intensity value 3 and pixel intensity value 1 adjacent to each other in 2 times. Hence, the matrices are symmetric in nature and the co-occurrence pairs

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

obtained by choosing  $\theta$  equal to  $0^\circ$  would be similar to those obtained by choosing  $\theta$  equal to  $180^\circ$ . This concept extends to  $45^\circ$ ,  $90^\circ$  and  $135^\circ$  as well.

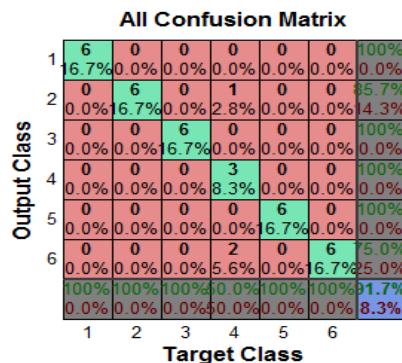
**F. Recognition system :** Once the features are extracted it is essential to recognize the currency based on these features by using an effective currency recognition system called classifier. One of the most common classifier which had been used recently is Artificial Neural Network.

**i. Training Phase:** Initially the feed forward back propagation neural network is trained with 36 samples of each currency. The performance and training state results are plotted and shown in the following figures. Feed forward back propagation neural network originate from the problem of interpolating a set of data in a hyperspace to find the best hyper-plane that interpolates that set of data. In feed forward back propagation neural networks, there are three layers: input layer, hidden layer, output layer. The input layer consists of the data (or the pattern) that is to be interpolated (or classified). The hidden layer provides a nonlinear transformation from the input layer space to the hidden layer space. It usually consists of high number of neurons. The output layer provides a linear transformation from the hidden layer space to output layer space. The network is trained with images which contain normal, noisy and tilted images.

**ii. Testing Phase:** Once the network has been trained on different currencies with their corresponding feature vectors along with target class and the trained network is validated against a new test image to check the performance of proposed system.

## IV. EXPERIMENTAL RESULTS

The features color, edge and texture of six denominations are extracted and they are applied to train the network by using NPR tool in Matlab and they are represented as shown in the Fig. 3 .



**Fig. 3 Confusion plot**

The confusion plot showing the percentage of correct classification of scanned denominations obtained using predefined neural network given in NPR tool.

Fig. 3 shows the recognition results. Average recognition was seen as 97.7% which is quite reasonable and acceptable in various cases. Plots the training, validation and test performances given the training record returned by the function train. Mean square error is the averaged square difference between outputs and targets. Lower values are better, zero means no error. The performance of network classifier is given by performance plot as shown in Fig. 4

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

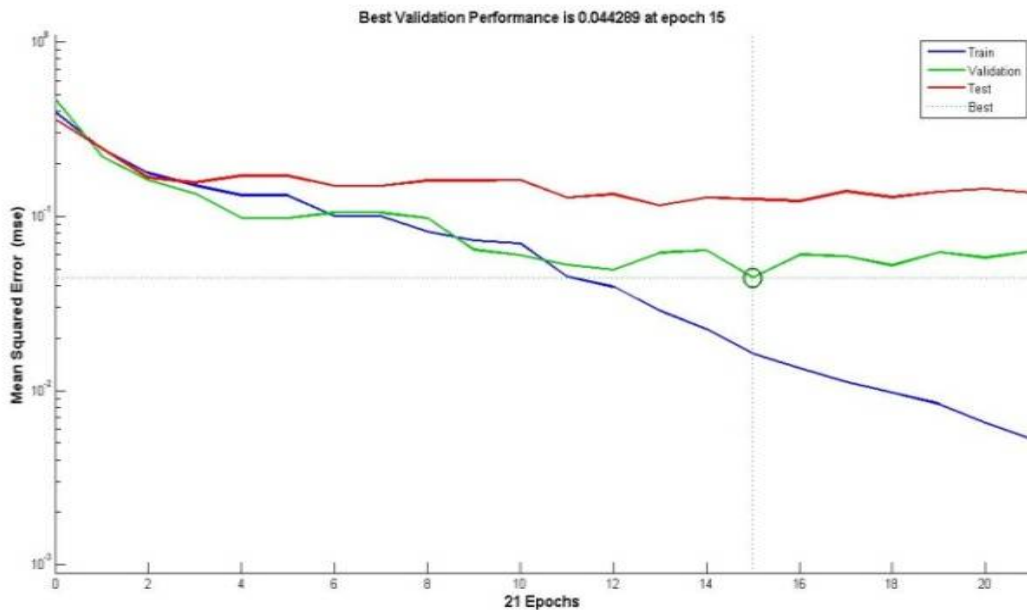


Fig.4 Performance of network classifier

Maximum percentage of correct classification of 91.7% was obtained with number of hidden nodes 13 as shown in Fig. 5

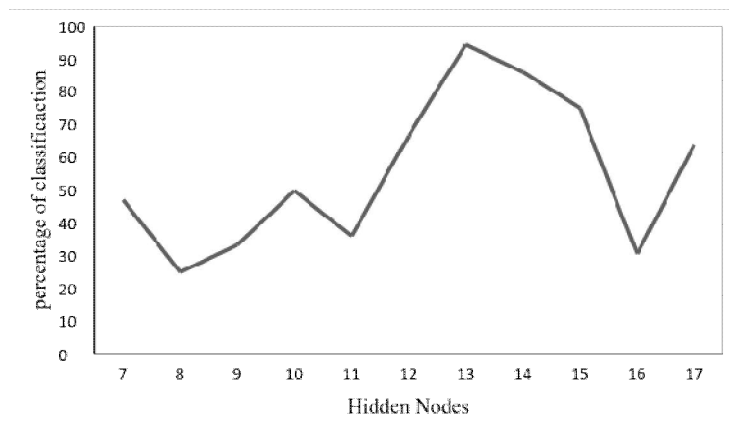


Fig.5 percentage of classification

## V. CONCLUSION

Paper currency recognition is an important application of pattern recognition. Many studies were made to recognize currencies using neural networks. In this project another method of recognizing currencies has been introduced, which is based on correlation between images. The method uses feed forward back propagation neural networks. This method is quite reasonable in terms of accuracy.



# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirset.com](http://www.ijirset.com)

Vol. 6, Issue 4, April 2017

## REFERENCES

- [1] Gonzalez RC, Woods RE. Digital Image Processing, Prentice Hall, 3<sup>rd</sup> edition; 2009.
- [2] Toussaint GT. Proximity graphs for nearest neighbor decision rules: Recent progress. Proceedings of the 34th Symposium on the INTERFACE, p. 17-20; 2002.
- [3] Vila A, Ferrer N, Mantecon J, Breton D, Garcia, JF. Development of a fast and non-destructive procedure original and fake euro notes.
- [4] Zhang EH, Jiang B, Duan JH, Bian ZZ. Research on paper currency recognition by neural networks. In Proceeding of the second international conference on machine learning and cybernetics, p. 2193– 2197; 200
- [5] Liu Q, Tang L. Study of Printing Identification Based on Multi-spectrum Imaging Analysis, Proceedings of the International Conference on Computer Science and Software Engineering, p. 229 – 232; 2008.S.Bhuvaneswari, T.S.Subashini, “Automatic Detection and Inpainting of Text Images”, International Journal of Computer Applications (0975 – 8887) Volume 61– No.7, 2013
- [6] Mark Nixon, Alberto Aguado. Feature Extraction & Image Processing, Academic Press, 2<sup>nd</sup> edition; 2008.