

# SOLVING LINEAR EQUATIONS FROM AN IMAGE USING ANN

Mithun Roy<sup>1</sup>, Anupam Mukherjee<sup>2</sup>, Alok Basu<sup>3</sup>, Pratik Kumer Halder<sup>4</sup>

<sup>1</sup>Assistant Professor, Computer Science & Engineering, Siliguri Institute of Technology, West Bengal, India

<sup>2</sup>Assistant Professor, Computer Science & Engineering, Siliguri Institute of Technology, West Bengal, India

<sup>3</sup>Assistant Professor, Computer Science & Engineering, Siliguri Institute of Technology, West Bengal, India

<sup>4</sup>Assistant Professor, Computer Science & Engineering, Gargi Memorial Institute of Technology, West Bengal, India

## Abstract

Optical character recognition has a great impact in image processing application. This paper combines the concept of OCR and feed-forward artificial neural network to solve mathematical linear equations. We implement blob analysis and feature extraction to extract the individual characters to a captured image which having some mathematical equations. We are constructing 39 character set which having some numbers, alphabet and operators. Training of these character set is done by using supervised learning rule. If that image satisfying linear equation condition then our proposed algorithm solve this equation and generate the output. This paper tries to increase the recognition rate more than 87%. The result achieved from the training and testing on the network of the letter recognition is satisfactory.

**Keywords:** Artificial Neural Network, Linear Equation, Recognized rate, Optical Character Recognition.

\*\*\*

## 1. INTRODUCTION

The objective of this paper is to capture an input text image which having an linear mathematical equations and take out those equations from that image and also extract the characters from that equations, characters are analyzed by comparing its features to distinguished each character from the input character sets.<sup>[2][4]</sup> We are focusing on those text images which are written in the following fonts, such as *Times New Roman*, *Arial Black*, *Cambria*. We are dealing with such characters which are normally used in linear equations, like digits (0-9), arithmetic operators (+, -) and assignment operator (=). Our input set contains 39 different characters and each size of the character is 7X5. So, Input matrix size will be  $35 \times 39$ . And target matrix size will be an identity matrix of order  $39 \times 39$ .<sup>[7][11]</sup> Each column identifies the different characters of the input matrix. These proposed methodologies also reduce the recognition problem and the computational time.

The overall optical character reorganization (OCR) is done in this paper by using Artificial Neural Network (ANN).<sup>[10]</sup> In this case we are used feed-forward neural network with multiple hidden layers. OCR is a very handful tools for image processing application to translate the optically scanned image to digital bitmap form.

### 1.1 Linear Equation

A **linear equation** is an algebraic equation in which each term is either a constant or the product of a constant and (the first power of) a single variable.<sup>[12]</sup>

Linear equations can have one or more variables. Linear equations occur abundantly in most subareas of mathematics and especially in applied mathematics.<sup>[12]</sup>

In mathematics, a **system of linear equations** (or **linear system**) is a collection of linear equations involving the same set of variables. For example,

A general system of  $m$  linear equations with  $n$  unknowns can be written as

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ \vdots & \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m \end{aligned} \quad (1)$$

Here  $x_1, x_2, \dots, x_n$  are the unknowns,  $a_{11}, a_{12}, \dots, a_{mn}$  are the coefficients of the system, and  $b_1, b_2, \dots, b_m$  are the constant terms.<sup>[12]</sup>

Often the coefficients and unknowns are real or complex numbers, but integers and rational numbers are also seen, as are polynomials and elements of an abstract algebraic structure. There are two general ways we are represents the system of linear equations.<sup>[12]</sup>

#### 1.1.1 Vector Equation

One extremely helpful view is that each unknown is a weight for a column vector in a linear combination.<sup>[12]</sup>

$$x_1 \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{m1} \end{bmatrix} + x_2 \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{m2} \end{bmatrix} + \cdots + x_n \begin{bmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{mn} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \quad (2)$$

### 1.1.2 Matrix Equation

The vector equation is equivalent to a matrix equation of the form  $AX = b$ .

where  $A$  is an  $m \times n$  matrix,  $X$  is a column vector with  $n$  entries, and  $b$  is a column vector with  $m$  entries.[12]

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \quad (3)$$

## 2. PROPOSED METHODOLOGY

In this paper the process of character recognition system is divided into four major categories.

1. Preprocessing and feature extraction.
2. Reorganization using feed-forward neural network.
3. Verification and validation of linear equations.
4. Solving equations and accuracy measurement.

### 2.1 Preprocessing and Feature Extraction:

In this phase we, prepare an input image which is captured through digital input media like scanner or digital camera/web camera. In this case we are trying to capture those images which are having linear mathematical equations.

### 2.2 Image Conversion:

It is the process that converts input RGB image to Gray scale image (an intermediate image). We need to do this conversion because we need binary image for further operations. In next step gray scale image is converted into binary image. Binary image is a digital image that has two possible values, either 0 or 1. It may be represented in either black or white colors. [1]

The flow diagram of our proposed model is given below:

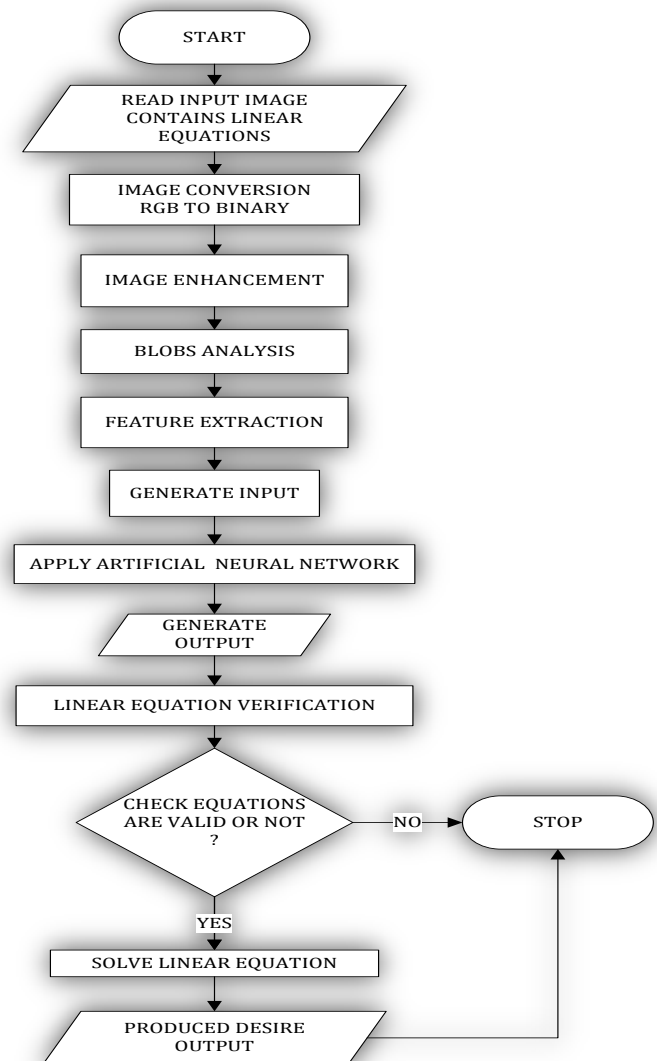


Fig.1. Proposed Model

### 2.3 Image Enhancement

It is the process that reduces noise and blurring effect of that input image. Median filter is used for noise reduction. After image filtering procedure edge detection technique is implemented. 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. [7][8][9]

### 2.4 Blobs Analysis

In Blobs analysis foreground and background objects are separated by using thresholding method. It is used to separate object and background, which is divided image into two modes.[3] The way to resolve both categories is by assigning a thresholding value  $T$ . [1][5][6] Each point  $(x, y)$  which have value  $f(x, y) \leq T$  is called foreground object,

and each point  $(x, y)$  which have value  $f(x, y) > T$  is called background object. A threshold image  $g(x, y)$  is defined as,

$$g(x, y) = \begin{cases} 1, & \text{if } g(x, y) \leq T \\ 0, & \text{if } g(x, y) > T \end{cases} \quad (4)$$

## 2.5 Feature Extraction

In feature extraction we are considering some steps for extracting a vector. We are focusing on content of the image not the total part of the image. Our main intention is finding a vector from the image. We have only two types of data on the image. Those are 1 for the white space and 0 for the black space. Now we have to pass the following steps for creating input vector for a particular character or image. Those are: 1. Pixel grabbing 2. Finding probability of making square.<sup>[1][11]</sup>

## 2.6 Generate Input Matrix:

After blobs analysis, objects are detected. Then we extract each object and resize it in a 7x5 dimensions. This information stored in a column vector which having 35 rows.

## 2.7 Artificial Neural Network:

Neural networks are a promising new generation of information processing system that demonstrate the ability to learn, recall and generalize from training patterns or data. ANNs are good at task such as pattern matching and classification, function approximation, vector quantization and data clustering. ANNs have a large number of highly interconnected processing elements (called nodes) that usually operate in parallel and are configured in regular architecture. The collective behavior of ANN, like a human brain, - demonstrate the ability to learn, recall and generalize from training patterns or data.<sup>[2][4]</sup>

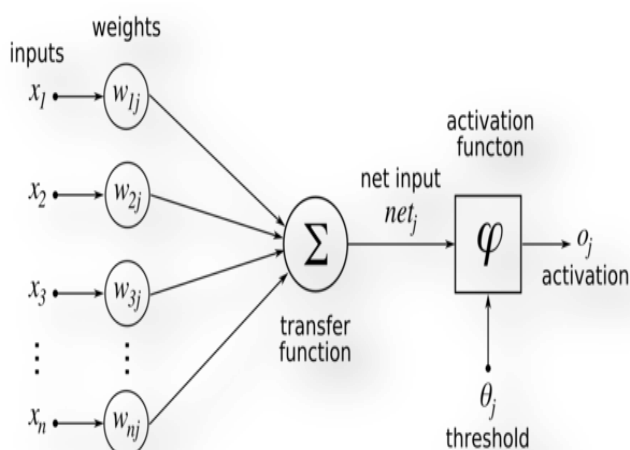


Fig.2. Mathematical model of ANN

The  $i^{\text{th}}$  processing element computes a weighted sum of its input and output.

$Y_i = 1$  (firing condition) or 0 (non firing condition)

According to whether this weighted input sum is above or below a certain threshold  $\theta_i$ .

$$Y_i(t+1) = a(\sum_{j=1}^m w_{ij}x_j(t) - \theta_i) \quad (5)$$

Where the activation function is a step function

As we know that general learning rules are classified into three categories in ANN.

1. Supervised learning rules.
2. Unsupervised learning rules.
3. Reinforcement.

We focused in supervised learning rules. In supervised learning rule for a particular input  $X$  a corresponding actual output will be generated and that value is checked with desire output, if it is matched then it is OK otherwise a error signal will be generated.<sup>[2][4]</sup>

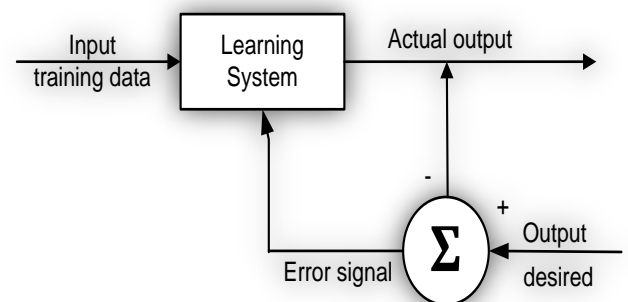


Fig.3. Supervised learning.

In this process we introduce the column vector as a input parameter to the processing element in artificial neural network (ANN). We are introducing multiple hidden layers in this case. Resultant value of this feed-forward ANN is checked with target matrix and the final output is extracted.

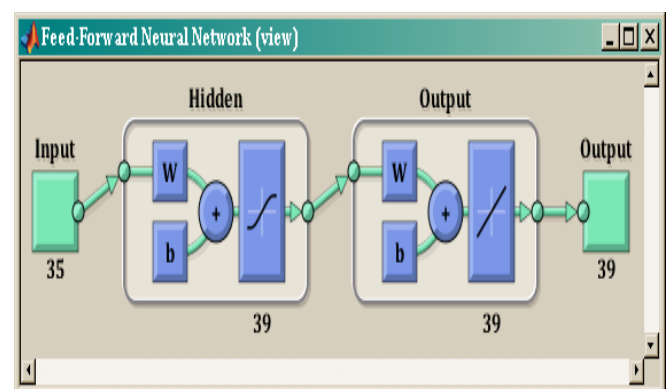


Fig.4. Feed-forward ANN

## 2.8 Liner Equation Extraction and Verification:

In this section after extracting each character from the processed image, we checked whether each character is recognized by this system correctly? If yes, then we use finite automata and check those extracted valid characters make linier equation or not. We check the Linier Equations are correct or not using the following Finite Automata.

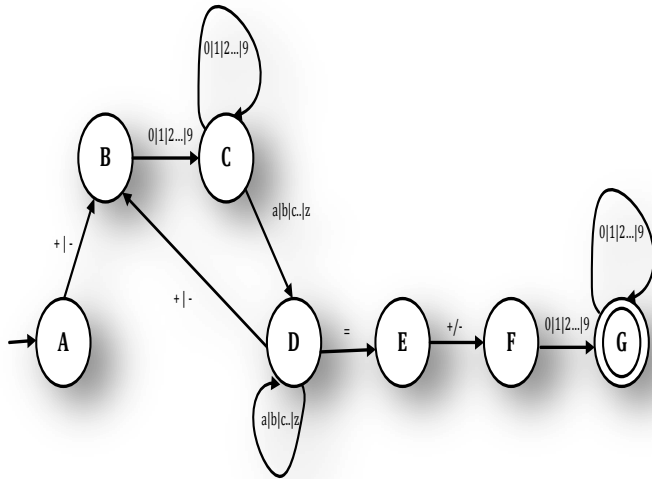


Fig.5. Linear Equation recognizer

### Acceptability of a string by Finite Automata:

A string  $W$  is accepted by a finite automata  $M = \{Q, \Sigma, \delta, q_0, F\}$  if  $\delta(q_0, W) = P$  for some  $P$  in  $F$ . This concludes that the string is accepted by the final state.

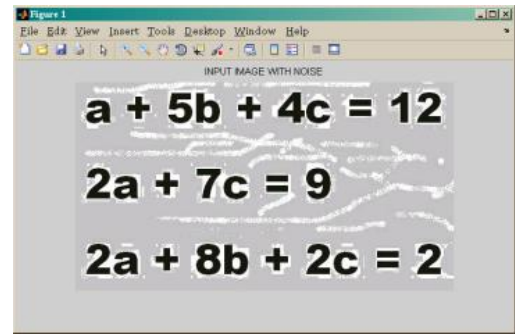
From the above finite automata  $M$  contains  $Q = \{A, B, C, D, E, F, G\}$ ,  $\Sigma = \{+, -, =, 0, 1, \dots, 9, a, b, \dots, z\}$ ,  $F = \{G\}$  and  $q_0 = \{A\}$  as the initial state.

e.g. let us say  $W = '2x + 34y - 290z = -49'$  be a input string the above finite automata checks the input sting is the valid linear equation or not.

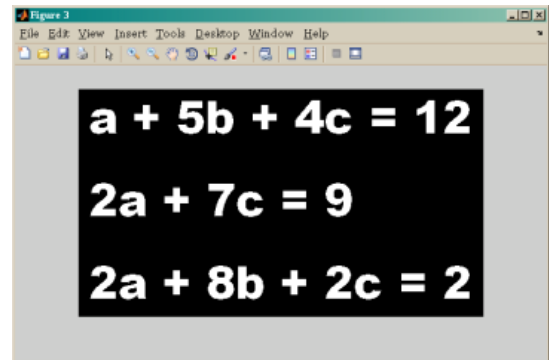
$\delta(A, 2x + 34y - 290z = -49)$   
 $= \delta(B, 2x + 34y - 290z = -49)$   
 $= \delta(C, x + 34y - 290z = -49)$   
 $= \delta(D, +34y - 290z = -49)$   
 $= \delta(B, 34y - 290z = -49)$   
 $= \delta(C, 4y - 290z = -49)$   
 $= \delta(C, y - 290z = -49)$   
 $= \delta(D, -290z = -49)$   
 $= \delta(B, 290z = -49)$   
 $= \delta(C, 90z = -49)$   
 $= \delta(C, 0z = -49)$   
 $= \delta(C, z = -49)$   
 $= \delta(D, = -49)$   
 $= \delta(E, -49)$   
 $= \delta(F, 49)$   
 $= \delta(G, 9)$   
 $= G \text{ (Accepted)}$

### 3. EXPERIMENTAL RESULT

Step I: Read input image from input devices (like scanner or digital camera). Then convert that input RGB image to gray scale image and remove noise using median filter, finally convert that gray scale image to binary image.



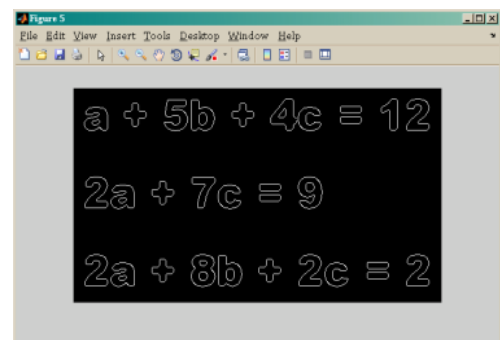
(a)



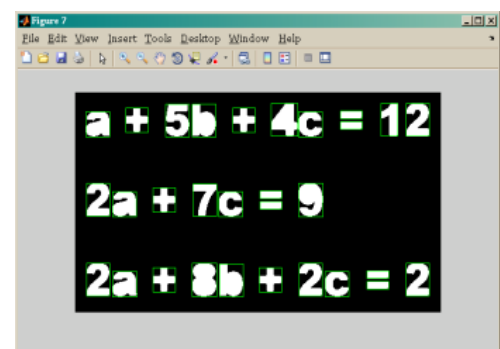
(b)

Fig.6. a. Input image (after scanning) b. Binary Image

Step II: After detecting binary image we are using sobel operator to generate edge detected image which is followed by image filling and blobs analysis.



(c)



(d)

Fig.7. c. Edge detected image d. Image Filling and Blobs analysis

Step III: Recognition of characters is done using feed-forward ANNs.

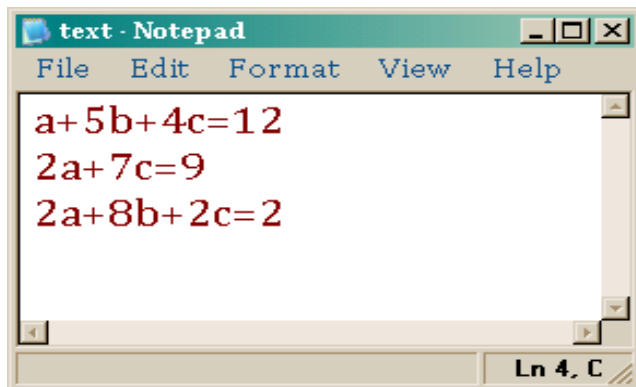


Fig.8. After Character Extractions & Verification

Step IV: After character extraction and verification of the extracted valid equations using regular expression we generate the resultant output which is shown below.

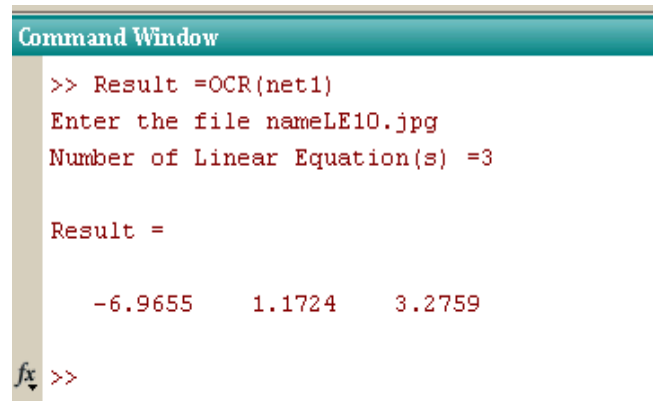


Fig.9. Resultant values of the following three linear equations

#### 4. RESULT ANALYSIS

The result achieved from the training and testing on the network of the letter recognition is at around 87%. Previously it is around 80-82%. We increase the recognition rate at around 5% more. In this part we had shown the details of the result analysis and network performance. We choose 39 input characters and each and individual character is checked and validate through our reorganization system. Those characters which are not recognized by the system are treated as an error which is marked by the “yellow” color. After actual verification and validation of characters we have got 87% success rate. <sup>[11]</sup>

IC: INPUT CHARACTER  
RC: RECOGNIZED CHARACTER  
RR: RECOGNIZED RATE

IC	RC	RR	IC	RC	RR	IC	RC	RR
1	1	> 87 %	a	a	> 87 %	n	n	> 87 %
2	2	> 87 %	b	b	> 87 %	o	0	0%
3	f	0%	c	c	> 87 %	p	p	> 87 %
4	4	> 87 %	d	d	> 87 %	q	a	0%
5	5	> 87 %	e	e	> 87 %	r	r	> 87 %
6	6	> 87 %	f	f	> 87 %	s	s	> 87 %
7	7	> 87 %	g	g	> 87 %	t	t	> 87 %
8	8	> 87 %	h	h	> 87 %	u	u	> 87 %
9	9	> 87 %	i	i	> 87 %	v	v	> 87 %
0	0	> 87 %	j	l	0%	w	w	> 87 %
+	+	> 87 %	k	k	> 87 %	x	x	> 87 %
-	-	> 87 %	l	l	> 87 %	y	v	0%
=	=	> 87 %	m	m	> 87 %	z	z	> 87 %

Recognized rate:  $(34/39) \times 100\% = 87\%$

#### Network Performance Analysis

The data set is checked by typical feed-forward neural network. We can check the network performance and determine if any changes need to be made to the training process, the network architecture, or the data sets. The network mean squared error performance was rated at 0.0013149 during epoch 10.

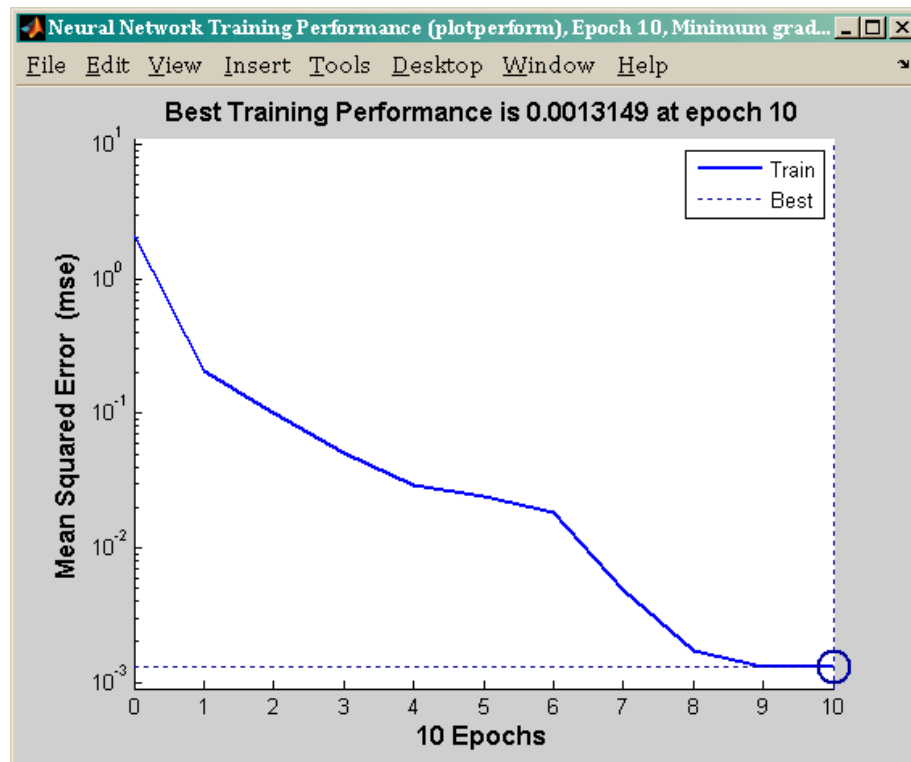


Fig.10. Plot-performance

Plot performs plot the training validation and test performances on the training record.

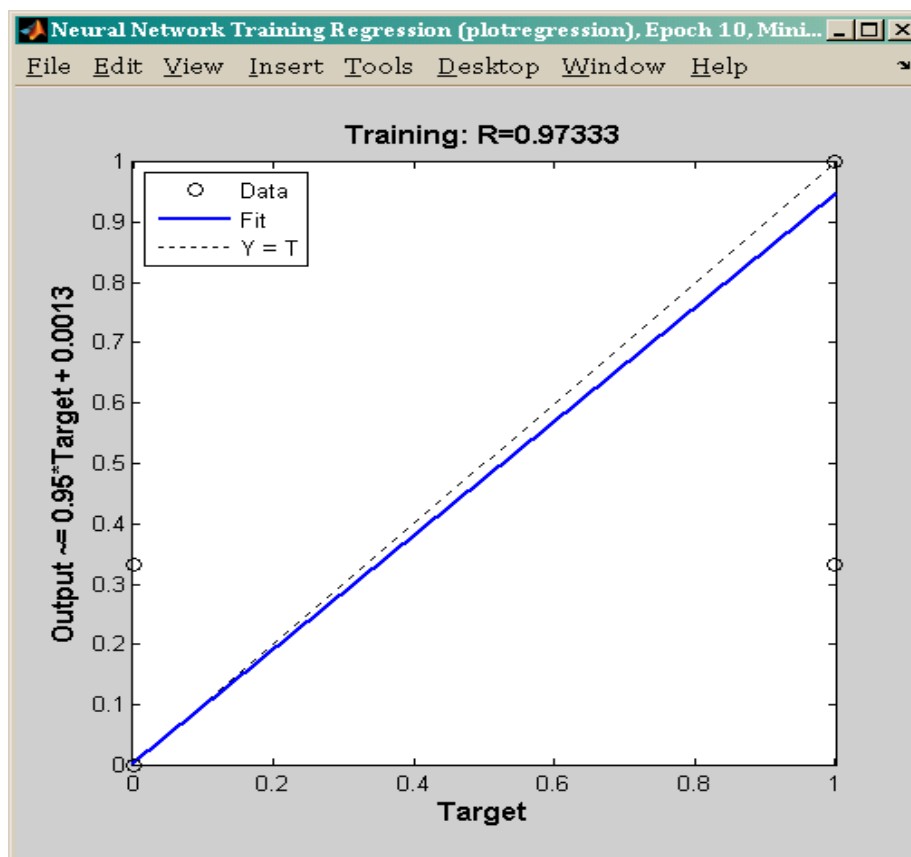


Fig.11. Plot-regression

Plot the linear equation of targets relative to outputs.



## 5. CONCLUSION

Most important part of this paper is its real life implementation. Any person can capture an image using image capturing devices from any books having system of linear equations, and then our proposed model is good enough to detect and generate the resultant output. In this paper we proposed a feed forward ANN based model which is used to recognize the character set which are involve in system of linear equations. We are not only deal with optical character recognition (OCR) but also proposed a model which can extract the features of the input character set. Verification and validation part is also equally important; acceptance of the string is verified by using finite automata. Solving system of linear equations is done using Gauss Elimination method. The result achieved from the training and testing on the network of the character recognition is at around 87%. Previously it is around 80-82%.<sup>[11]</sup> We increase the recognition rate at around 5% more.

## ACKNOWLEDGEMENTS

We tender our heartily gratitude to our parents, family members and friends.

## REFERENCES

- [1]. Gonzalez, Rafael C., Richard E. Woods. :” Digital Image Processing.” Ed III, Pearson Education Asia.
- [2]. Ruck, D.W. Rogers, S.K. and Kabrisky, M. “Feature Selection Using a MultiLayer Perceptron”. In J. Neural Network Comput. 2 (2):40-48, 1990.
- [3]. S.Arya and D.M.Mount. “Approximate nearest neighbor searching”. Proceeding 4th Annual ACM-SIAM Symposium on Discrete Algorithms, pages 271-280, 1993.
- [4]. Rossi, F.”Attribute Suppression with Multi-Layer Perceptron”. In Proceedings of IEEE-IMACS’96, Lille, France. 1996.
- [5]. D.Comaniciu and P.Meer. “Robust analysis of feature spaces: color image segmentation”. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pages 750-755, 1997.
- [6]. J.Shi and J.Malik. “Normalized cuts and image segmentation”. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 731-737, 1997.
- [7]. Viola, P. and Jones, M. Robust Real Time Object Detection, In International Journal of Computer Vision, 1(2), 2002.
- [8]. Jinshan Tang Eli Peli, and Scott Acton “Image Enhancement Using a Contrast Measure in the Compressed Domain”, IEEE Signal processing Letters , Vol. 10, NO. 10, October 2003.
- [9]. S. S. Agaian, S. Blair and K. A. Panetta, “Transform coefficient histogram-based image enhancement algorithms using contrast entropy”, IEEE Transaction on Image Processing. vol. 16, no. 3, pages. 741-758, 2007.
- [10]. A. A. Wadud, M. Kabir, M. H. Dewan and M. C. Oksam, “A dynamic histogram equalization for image contrast enhancement”, IEEE Trans. Consumer Electronic , vol. 53, no. 2, pages 593-600, 2007.

[11]. Md. Alamgir Badsha, Md. Akkas Ali, Dr. Kaushik Deb, Md. Nuruzzaman Bhuiyan, “Handwritten Bangla Character Recognition Using Neural Network”, International Journal of Advanced Research in Computer Science and Software Engineering”. Vol 2, Issue 11, pages. 307 – 312, November 2012.

[12]. Hamming, R.W., "Introduction to Applied Numerical Analysis", McGraw-Hill Book Co., Inc., New York, San Francisco, Toronto, London. 1971

## BIOGRAPHIES



Mithun Roy has obtained M.Tech. in Computer Science from West Bengal University of Technology. Presently he is working as an Assistant Professor in the department of CSE in Siliguri Institute of Technology. His research interest is in Digital Image Processing, Social Networking and Pattern recognition.



Anupam Mukherjee has obtained B.Tech. and M.Tech. in Computer Science from West Bengal University of Technology. Presently he is working as an Assistant Professor in the dept. of CSE in Siliguri Institute of Technology. His research interest is in Digital Image Processing, Computer Networking, Cryptography and Network Security.



Alok Basu has started his career as a Mechanical Engineer, at Alloy Steels Plant, SAIL in 1983. He worked there in different sections of his field in various capacities. Later he got a scope to be trained in COBOL and to work with the in-house software development and maintenance team for more than seven years. He was mainly involved in the development of Production Information & Order Servicing System in COBOL on MV 15000 (AOS/VS-II) platform. He joined in Siliguri Institute of Technology after completion of M. Tech in Information Technology. He is serving this institute for more than 10 years and presently holding the chair of HOD.



Pratik Kumer Halder has obtained B.Tech. and M.Tech. in Computer Science from West Bengal University of Technology. Presently he is working as an Assistant Professor in the dept. of CSE in Gargi Memorial Institute of Technology. His research interest is in Digital Image Processing, visual cryptography.