OPTICAL CHARACTER RECOGNITION
FOR
PRINTED TEXT

A PROJECT REPORT SUBMITTED
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF

MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE

By

M.Supriya and K.N.V.L. Subrahmanyeswari

DEPARTMENT OF COMPUTER/INFORMATION SCIENCES
SCHOOL OF MATHEMATICS AND COMPUTER/INFORMATION SCIENCES
UNIVERSITY OF HYDERABAD
HYDERABAD – 500 046, INDIA

1999

CERTIFICATE

This is to certify that the thesis entitled "Optical Character Recognition for Printed Text" being submitted by M.Supriya and K.N.V.L.Subrahmanyeswari in partial fulfillment for the award of the degree of Master of Technology in Computer Science, is a record of the bonafied work carried out by them under my supervision at the University of Hyderabad.

The matter embodied in this project report has not been submitted to any other university for the award of any other degree or diploma

Prof. Arun Agarwal
Head
Dept. of CIS
University of Hyderabad

Dr Arun K. Pujari
Professor, Supervisor
Dept of CIS
University of Hyderabad

Prof. K. Vishwanath
Dean
School of Mathematics and Computer/Information Sciences,

Acknowledgements

We take this opportunity to express our profound sense gratitude to our guide Prof. Arun K. Pujari, for his inspiring guidance. We would like to thank him for the invaluable suggestions, encouragement, advice and timely help he provided.

We would like to sincerely thank Mr. M. Srinivasa Rao for providing the Neural Network used for our recognition mechanism.

We expess our heartfelt gratitude to the staff of A.I. Lab for their valuable support and cooperation throughout.

We also thank our classmates Seema and Shazia for their understanding support.

Last, but not the least, we thank our juniors for their encouragement and cheerful company.

Supriya Madamanchi

K.N.V.L.Subrahmanyeswari

ABSTRACT

Making a computer perform the task of analyzing and understanding a document is a challenging and interesting project Our project tries to recognize the text characters of a printed document

The report describes the development and implementation of a Text Recognition System In our present work we assume the existence of a front end video scanner, the input then is a digitised bilevel image encoded in the form of a bitmap

The project is divided into two parts preprocessing and recognition. In the preprocessing phase the document which is fed as a bitmap image is analyzed and lines are extracted. These lines are then further processed and the words and letters in each word are obtained.

The next phase, i.e., the recognition phase, uses a Neural Network to recognize the letters The standard patterns for the network are obtained from the scanned image of the reference set of the same font and style as the document to be recognised. The letter from the document are then fed as the test patterns to Neural Network which outputs the recognised characters These characters are then grouped in to lines and words and are printed as a document

The project is implemented using Visual C++ on windows 95 platform.

CONTENTS

Chapter 1

INTRODUCTION

1.1 Overview

The advance of office automation and the onset of the information age is fundamentally grounded in the ability to create, manipulate, store, retrieve and transmit electronic documents Word processing applications allow us to easily create and edit electronic documents and compression of the documents' electronic representation allows efficient storage and transmission, the text analysis community has focussed on the processing, indexing and retrieval of text from large repositories For documents whose entire life cycle is electronic, these tasks are relatively straightforward and users have therefore come to view documents as manipulable, searchable entities Unfortunately, for those documents that were created manually, or for those documents whose electronic representation is simply not available, systems must deal with scanned images of the hard copy representations

To overcome this enormous and ever increasing problem, there is a need for the realisation of computer based systems which are capable of providing automated analysis and interpretation of paper-based documents

The move from paper-based documentation to paper-less storage and retrieval system has been prompted by advantages such as efficiency of storage and retrieval, high saving of storage space, ability to update and revise the document stored with minimum effort etc

The ultimate goal of developing reading machines, which copes with the enormous flood of paper documents such as bank cheques, commercial forms, government records, mail sorting etc., generated by the expanding technological society, has been the principal motivation for OCR systems Character recognition plays a significant role in document analysis systems. Today's Optical Character Recognition Systems "read" text by scanning an image of the document and then converting it to characters, letters, numbers, and symbols that the computer can manipulate. Once the document is in character form, the computer can be used to reformat the text in the document, do repetitive mathematical calculations or sort information in the database.

Methodologies in character recognition:

The methodologies applied in character recognition are not different from those applied in any general image or pattern recognition problem, with respect to image analysis and pattern recognition. However, because of the particularity of the character recognition problem, the methodologies can be summarized according to Gaillat and Berthod as follows.

(i) *Point by point global comparison.* This process involves a point by point comparison of all the pixels that are stored with the pixels of the image.

(ii) *Global transformations.* For example, Karhunen-Loeve, Fourier moment calculations and finally rotation according to the principal axes of inertia.

(iii) *Extraction of the local properties.* Such local properties are the end points, the pronounced angles or corners, the T-junctions, and the crossing of strokes. This extraction can be achieved by various methods, e.g., by means of fixed masks of variable positions. It is often prerequisite to thin the character and then to apply the above methods of extraction.

(iv) *Search for the extraction with certain lines.* The lines can either form rectangles or simply be vertical or horizontal.

(v) *Analysis by means of curvatures.* This involves curve following, detection of concavities and geometric analysis.

(vi) *Structural methods:* Such methods include decomposition of the character into its constituent elements, the topological description and the reduction character into a graph.

1.2 Historical Background

One of the earliest attempts in character recognition was that of Grimsdale et al [Grimsdale et al , 1958] wherein, in a pioneering work, they described a method where the input pattern is scanned by a flying spot scanner Subsequently it is analyzed for shape by a digital computer, which extracts the basic features These features are compared to values that have been stored in the computer, in order to reach the proper decision It was envisaged that for hand written characters to be recognized the same approach would be adequate but with a higher degree of complexity Another work for that period was that of Bledsoe and Browning [Bledsoe and Browning, 1959] who described a rather more general approach using a photo mosaic of 10 x 15 photocells

In the early 1960's Murray Eden at M I T put forward the idea that all Latin script characters can be formed by 18 strokes, which in turn can be generated from a subset of four strokes, called segments This concept was the origin of a great deal of research at M I T and elsewhere and is known as the "analysis-by-synthesis" method, as for examples see those by Blesser, Cox, Shillman, and finally yoshinda in Chinese characters But the great importance of Eden's work was that he formally proved that all hand written characters are formed by a finite number of schematic features, a point that was implicitly included in previous works This notion was later used in all methods in syntactic (structural) pattern recognition applied to character recognition.

In Casey and Nagy [Casey and Nagy, 1966] at I B M presented one of the first attempts in Chinese character recognition Since the Chinese alphabet was about 1000 characters a step-by-step approach was applied That is, they first grouped similar characters, and then "group masks" and then finally "individual masks" were employed However, later works showed that a number of strokes and their positions could be adequate for recognition of Chinese characters In 1968, Casey and Nagy introduced an unconventional approach to character recognition, the so-called "autonomous reading machine", since no previous training or *a Priori* information about the character was needed, but instead the known letter pair frequencies of the language were used for recognition

Parks et al [Parks et al ,1974] produced a feature extraction method for extracting lists of topological features and a hierarchically structured articulate method of recognition Tou and Gonzalez [Tou and Gonzalez , 1972] introduced a two stage scheme wherein the first stage measurements are taken by means of a measurement grid in order to separate the pattern classes into several subgroups, and in the second stage a number of specialized features are extracted

Pavlidis [Pavlidis, 1975] making use of his split and merge algorithm, produced polygonal approximations of the characters which could provide enough information for both the character shapes and the syntactic analyzer

1.3 Applications Of OCR

Apart from being used as document storage systems which eliminate laborious and repetitive work needed to index documents, OCR systems have many other applications A few of them are

- Use in postal department - for postal address reading and as reader for hand-written and printed codes
- Use by blind people - as a reading aid when synthesized with the speech system
- In automated Cartography, information units and libraries for facsimile
- For business applications – financial business applications like check sorting strategy optimization.

- In law enforcement applications, in educational administrations examination assessment and as attendance record evaluation
- For digital bar code reading and as hand writing analyzer – for automatic writer recognition and signature verification
- Use in customer billing as in telephone billing system.
- Use as a recognition of signatures in credit or personal identification system

1.4 Organization Of The Thesis

Chapter 1 deals with the general concepts and applications of Optical Character Recognition.

Chapter 2 deals with Methodologies used for OCR and describes the NN used for recognition.

Chapter 3 explains the process of recognition and describes the various function and modules implemented in the present work.

Chapter 4 gives an account of the tests conducted and presents the details of the results obtained thence.

Chapter 5 presents the conclusions drawn along with the limitations of the system as well as scope for future work.

Chapter 2

CHARACTER RECOGNITION METHODOLOGIES

2.1 Survey Of Previous Work

Literature on OCR may be primarily classified into two categories based on the two approaches for recognition 1) feature mapped and 2) image mapped systems as mentioned earlier Some of the literature on OCR includes surveys, practical working implementations and theories for OCR We begin the literature survey with a discussion of some OCR systems

2.1.1 Feature mapped systems

Pavlidis [Pavlidis, 1986] described a vectorizer and feature extractor for document recognition He described a thinning algorithm that operates directly on the run length encoding of a binary image The algorithm finds ready use in applications ranging from Cartography to character recognition Strokes, arcs, holes endpoints etc are found by this algorithm The performances of the algorithm for both character (English and Arabic numerals) and document recognition have been illustrated [Pavlidis, 1986]

Sinha et al [1979] designed a syntactic pattern analysis system with embedded Picture Language called PLANG for the recognition of Devanagiri script It assumed a rectangular picture frame for every picture It consisted of primitives, partitioning functions, composed macros and frame functions Structural descriptions of each symbol of the script were stored in terms of primitives and their relationships Recognition involved a search for primitives on the preprocessed and labeled patterns based on stored descriptions First a word in Devanagiri script was segmented into composite characters, the composite characters were decomposed into meaningful parts and the decomposed parts then recognized Contextual information was used during recognition Experiments for recognition were performed for typed as well as constrained hand written characters

Sinha et al [Sinha et al 1987] proposed a rule based contextual post-processing scheme for Devanagiri text recognition in which the spatial relationships between the constituent symbols were used Word composition was checked for syntactical correction and, if found to be incorrect, symbols were substituted with their resembling counterparts An example of word composition rule is a vowel modifier symbol could appear only on a consonant

Yeh et al [Yeh et al, 1987] proposed a strategy to locate addresses on envelopes The envelope on which the address was to be located was segmented into regions of consistent print style and a region most likely to be the address chosen on grounds of position, size, print style etc without attempting to read the characters

Kimura et al [Kimura et al, 1991] combined two pattern recognition algorithms for recognition of unconstrained, isolated hand written numerals First, the features in the digitized character images were extracted Two features were considered of which one was a direction vector feature wherein a local histogram of chain codes was calculated and the feature vector was composed of these local histograms The feature vector had 64 components where 16 zones of the character image were included The second kind of features was the profile features, derived from profiles of external contours of the numerals These were character widths, ratio, location of extrema, and discontinuities in character profiles Among the two algorithms for recognition, the first algorithm used a modified quadratic discriminant function (MQDF), using direction-sensitive spatial features of the numerical image It required less computation time and storage than the ordinary quadratic discriminant function (QDF) and was less sensitive to estimation error in the covariance matrix It was reasonably well used in the case in which sufficient number of samples was not available The MQDF was derived from Bayes' decision rule for unknown Gaussian distribution under some assumption This statistical classification scheme was called the k-algorithm The second algorithm used features derived from the profile of the character in a structural configuration to recognize numerals A group was made up of Boolean expressions consisting of primitives for each numeral sub class These groups were defined as a set of primitives characterizing a specific sub class of a numeral Groups were determined by the similarity of their right and left profiles and were then combined to firm a chain for each numeral A chain is either a single group or a union of groups, all identifying a specific numeral recognition procedure A tree structure was utilized to determine the specific group to which a given test numeral belonged This structural classification was called as s-algorithm Results indicated low error and rejection rates for these algorithms and algorithms obtained by parallel and sequential combination of these two algorithms in different ways

Buchowski et al [Buchowski et al , 1988] have published a technical report describing an OCR system which can recognize the ten numerals printed in variety of fonts, styles and sizes The segmentation of the digitized image was done by first preprocessing the digital image The threshold of the image was determined using an automatic method from the image histogram by obtaining the largest peak of intensity, I and the peak with an intensity no greater than I-d for some d>0 The threshold value was the average of the intensities of these to peaks Now, from the binary image, lines of text were located from a horizontal projection of the image From the bare areas of projection, lines of text and isolated characters were identified and extracted, This character image was then thinned using four and eight connected regions Spurs obtained after thinning were removed and next, features were detected from the thinned image by locating fore ground components, holes, branch points, end points, vertical crossings and the ratio of a character's height to its width in nine zones of the character image Classification was done by traversing along a decision tree A branch test is associated with each non-terminal node of the decision tree which depended on variables (e g , number of components, holes, branch points, end points, vertical crossings), are predicates (e g , component crossing a zone, hole crossing location zone, branch point or end point occurrence in a zone) A 97% recognition performance was claimed for this system

Li et al [Li et al , 1992] used moment invariants as feature inputs to Neural Networks using an improved version of the back propagation algorithm for classification of some English character patterns Instead of using random weight and biases for initialization, they used a Correlation Based Learning Vector Analysis (CBLVA) algorithm, which calculated the initial parameters for the hidden layer by analyzing correlation values among training patterns A set of seven invariant moments, representing the shape characteristics of printed characters, were used as inputs for the Neural Network Experimental results show that this neural-net classifier converged much faster in training and achieves a classification-accuracy comparable to that of a standard propagation classifier

2.1.2 Image Mapped systems

Burr [Burr, 1988] performed some experiments on neural net recognition of spoken and written English text He describes a multilayer Neural Network with backpropagation training The coding of input signals was by means of a seven or thirteen segment LED readout Each bar was assumed to be a light detector with then shadow projecting the characters onto the set of segments Shadow codes were presented to for training and recognition He concluded that neural net performance on hand printed text was essentially the same for the nearest neighbor classifier (by measuring the Euclidean distance between a test vector and ten reference vectors and taking the reference having the smallest distance to a test vector) trained on the same data In the multilayer Neural Network training, the number of hidden units was varied and it was reported that with more hidden units, the network converged in fewer steps Network performance optimization was studied by 1) biasing input values to the high learning rate portion of the input neuron, and 2) eliminating the weights whose absolute values were less than a preset threshold In the first case, it was found that when the average value of the inputs over the entire training set was subtracted, learning was most rapid In the second case, a third of the near-zero output weights could be eliminated without adversely affecting the recognition accuracy, showing that the output neurons performed like logic gates

Fukushima et al [Fukushima et al , 1991] proposed a Neural Network model of visual pattern recognition, called the neocognitron for deformation invariant recognition purposes The neocognitron was a hierarchical network consisting of several layers of interconnected neuron-like cells The lowest stage of network was a two-dimensional input layer Each succeeding stage had a layer consisting of cells, called S (simple) cells, followed by another layer of cells, called the C (complex) cells S cells extracted features and C cells allowed for positional errors in features Learning could be supervised or unsupervised Different training patterns were used to train different layers of the network, with various aspects of deformation incorporated in the patterns Thus, deformation invariant pattern recognition was achieved

Perantonis et al [perantonis et al , 1992] discussed the classification and recognition of two-dimensional patterns independently of their size, position, orientation (termed transformation), local distortion and noise in the patterns by using high order networks A third order network was chosen to handle output invariants under translation, rotation and scaling transformations simultaneously Transformed, distorted and noisy hand written and typed numerals were presented to the network for testing its performance To obtain a third order network for example, all weights were set equal for which the line segments could be transformed into each other through translation, when these segments were equal in length and parallel to each other To build this into the network p-tuples of points in the plane were assigned to equivalence classes Thus, neuron activation functions were obtained such that a single layered feed forward network could classify images representing invariant features Bias terms are subsequent layers of weights did not affect the invariance properties of the network Zernike moments of test patterns were also computed and fed to a first ordered multilayered feed-forward Neural Network with backpropagation training as a classifier and this was compared with high order network recognition It was found that third-order networks were superior to the use of Zernike moments followed by a conventional Neural Network classifier in invariant pattern recognition tasks including recognition of distorted patterns, but not patterns corrupted by randomly distributed noise In both cases, the computational overhead incurred in coding the images was much less than the time required to train the classification network.

2.2 Overview of Present Work

The aim of the present work is to recognize printed English text using Artificial Neural Networks (ANNs) Earlier work on character recognition has been done using conventional pattern recognition techniques We make an initial attempt here of using Neural Networks for recognition with the aim of improving upon earlier methods which do not perform effectively in the presence of noise and distortion in the characters. Due to limitations of the Hopfield of Neural Network, a new scheme named as the Multiple Neural Network Associative Memory (MNNAM),as proposed by M B Sukhaswami et al was used The limitation in storage capacity has been overcome by combining multiple neural networks that work in parallel The Neural Network was designed and implemented by M Srinivasa Rao under the guidance of Prof Arun K Pujari , Dept of Computer Science, University of Hyderabad A brief description of the Neural Network is given in the following sections

2.3 The Neural Network

Artificial Neural Networks are abstract representations of brain information processing The hope to reproduce at least some of the flexibility and power of human brain by artificial means has lead to the subject of study known as Neural Networks, Neural computation or Brain like computation [Anderson 92]

2.3.1 Basic Concepts of ANN

The potential of Artificial Neural Networks (ANN) relies on massively parallel architecture composed of many simple computational elements connected by edges called weights The basic computational element in an Artificial Neural Network is called neuron and is also known as node or processing element Brain researchers have identified over 100 different kinds of biological neurons Processing elements also come in a variety of types McCulloch and Pitts [McCulloch 43] proposed a binary threshold unit as computational model for a neuron As a unit, the Neural Network can be represented with threshold and weight functions

In an Artificial Neural Network, inputs are fed to neurons through synapses (connection weights) Basically, the output of a neuron in a Neural Network is a weighted sum of its inputs, but a threshold function is also used to determine the final value or the output

The state of a neuron is nothing but the activity of a neuron at any time instance This attribute may be discrete or continuous valued An update rule or transition rule or activation rule is a rule for evaluating the state of one or more neurons at any time instance This attribute may be discrete or continuous valued An update rule or transition rule or activation rule is a rule for evaluating the state of one or more neurons under the existing conditions and changing them if necessary

Topology of ANN is the pattern of connectivity in such a way that it can be viewed as weighted directed graph in which Artificial neurons are nodes and directed edges (with weights) are connections between neuron outputs and neuron inputs Basing on the connection pattern (architecture), ANNs may be grouped into two categories Feed-Forward networks (non-recurrent) are those in which graphs have no loops, while Feed-Back (recurrent) networks are those in which loops occur because of feedback connections Broadly speaking feed forward networks are static, where as recurrent or feedback networks are dynamic systems When a new input pattern is presented the neuron outputs are computed Because of the feedback paths, the inputs to each neuron are then modified, which leads the network to enter a new state.

The state of all the neurons in the feedback network at any instant of time is the state of the neural network Stable states of a feedback network are the states of the network which do not change under usual disturbances in the states of the neurons of the network **Learning** in a neural network is the process of making certain set of states of network as stable states The function in which one can substitute the state values of the neurons that represents the energy of the network (at that instance) is called its energy function During the updation of the network, the value of the energy function decreases and eventually reaches a minimum In view of the above definition this state of the network is referred to as stable state and this minimum is called local minimum When this state is reached, the network is said to be stable

Learning Rules

The ability to learn is a fundamental trait of intelligence ANNs' ability to automatically learn from examples makes them attractive and exciting A learning process in the ANN context can be viewed as the problem of updating network architecture and connection weights so that a network can efficiently perform a specific task

A learning algorithm refers to a procedure in which learning rules are used for adjusting the weights There are three main learning paradigms *supervised*, *unsupervised* and *hybrid*

In *supervised learning* or learning with a "teacher", the network is provided with a correct answer (output) for every input pattern Weights are determined to allow the network to produce answers as close as possible to the known correct answers

In contrast, *unsupervised learning*, or learning without a teacher, does not require a correct answer associated with each input pattern in the training data set It explores the underlying structure in the data, or correlates patterns in the training data set It explores the underlying structure in the data, or correlates patterns in the data and organizes patterns into categories from these correlations

Hybrid learning combines *supervised* and *unsupervised* learning

There are four basic types of learning rules error-correction, Boltzmann, competitive learning and Hebbian learning A brief description of each of these is given below

Error-correction learning: In this learning paradigm, the network is given a desired output pattern for each input pattern During the learning process the actual output y generated by the network may not be equal to the desired output d The basic principle of error-correction learning rules is to use the error signal $(d-y)$ to modify the connection weights to gradually reduce this error

Boltzmann learning: It is a stochastic learning rule derived from information theoretic and thermodynamic principles The objective of Bolzmann learning is to adjust the connection weights so that the states of the visible units satisfy a particular desired probability distribution

Competitive learning: In this method of learning output units compete among themselves for activation As a result only one output unit is active at any given time This phenomenon is known as winner-take-all Competitive learning is found to exist in biological Neural Network's [Haykin, 1994]

Hebbian learning: The oldest learning rule is Hebb's postulate of learning [Hebb, 1949] Hebb's rule is based on the following observation from neuro-biological experiments if neurons on both sides of a synapse are activated synchronously, and repeatedly, the synapse's strength is selectively increased Mathematically, the Hebbian rule can be described as

$$w_{ij}(\tau + 1) = w_{ij}(\tau) + \eta y_j(\tau)\, x_i(\tau)$$

where x_i and y_j are the output values of neurons i and j, respectively, which are connected by synapse w_{ij}, and η is the learning rate Note that x_i is the input to the synapse The important property of this rule is that learning is done locally, that is, the change in the synaptic weight depends only on the activities of the two neurons connected by it This significantly simplifies the complexity of the learning circuit in a VLSI implementation

2.3.2 Hopfield Network

Many models of Neural Networks have been proposed to solve the problems of pattern recognition, prediction, optimization, associative memory and control And Hopfield model of Neural Network with associative recall is one of most suitable models to accomplish the present task

Hopfield [Hopfield 1982,1984] used the network energy function as a tool for designing recurrent networks and for understanding their dynamic behavior Hopfield's formulation made explicit the principle of storing information as dynamically stable attractors and popularized the use of recurrent networks for associative memory and for solving combinatorial optimization problems A Hopfield network with n units has two versions binary and continuous valued Let v_i be the state or output of the ith neuron. For binary networks v_i is 1 or 0 or for bipolar v_i is +1 or −1 and for continuous networks v_i could be any real value Let w_{ij} be the synaptic weight connecting neurons i and j the network dynamics for the binary Hopfield network are as follows

If the state of the ith neuron at time τ is denoted by $x_i(\tau)$, then the neuron at the next time step $\tau + 1$ is computed as

$$x_i(\tau+1) = sgn\left(\sum_{j=1}^{n} w_{ij}\, x_j(\tau) - t_i\right)$$

where $sgn(x)$ is a function that produces 1 if $x>= 0$ and 0 otherwise The central feature of the Hopfield network is that each state can be associated with a quantity called energy E The energy of the network at a particular state is given by [Hopfield, 1982]

$$E = -\tfrac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}\, x_i x_j + \sum_{i=1}^{n} t_i x_i$$

The energy function is of Lyapnov type which maps system state variables to real numbers and monotonically decreases with time Kosko [kosko, 1992] The central property of the energy function is that ,as network state evolves according to the network dynamics, the network energy always decreases and eventually reaches a local minimum point (attractor) where the network always decreases and eventually reaches a local minimum point (attractor) where the network stays with a constant energy

2.3.3 Associative Memory

When a set of patterns is stored as local minima (attractors or stable states) in a network, it can be used as an associative memory. Any pattern present in the basin of attraction of stored pattern can be used as index to retrieve it. The set of all initial state vectors that converges to a stable state is called its basin of attraction. An associative memory usually operates in two phases: memory storage and information retrieval. In the storage phase, the weights in the network are determined (using a learning rule like Hebb's rule) so that the attractors of the network memorize a set of patterns to be stored. In the retrieval phase, the input pattern is used as initial state of the network, and the network evolves according to its dynamics. A pattern is produced or retrieved when the network reaches equilibrium. The number of patterns stored in a network is called *capacity of the network*. It is infinite because a network with n binary neurons has a maximum of 2^n distinct states and not all of them are attractors. Some attractors are called spurious attractors (or spurious states) when they store patterns different from those in the training input.

The Hopfield network with Hebb's training rule suffers from the presence of spurious states, as well as with low storage capacity. Hence a new Neural Network architecture (MNN) consisting of set of set of Hopfield model connected in hierarchical pattern is proposed. Each Hopfield network is trained is trained by a new training scheme which avoids any spurious states is robust for retrieval with limited capacity. The limitation in capacity is overcome by the hierarchical structure of multi level Neural Network.

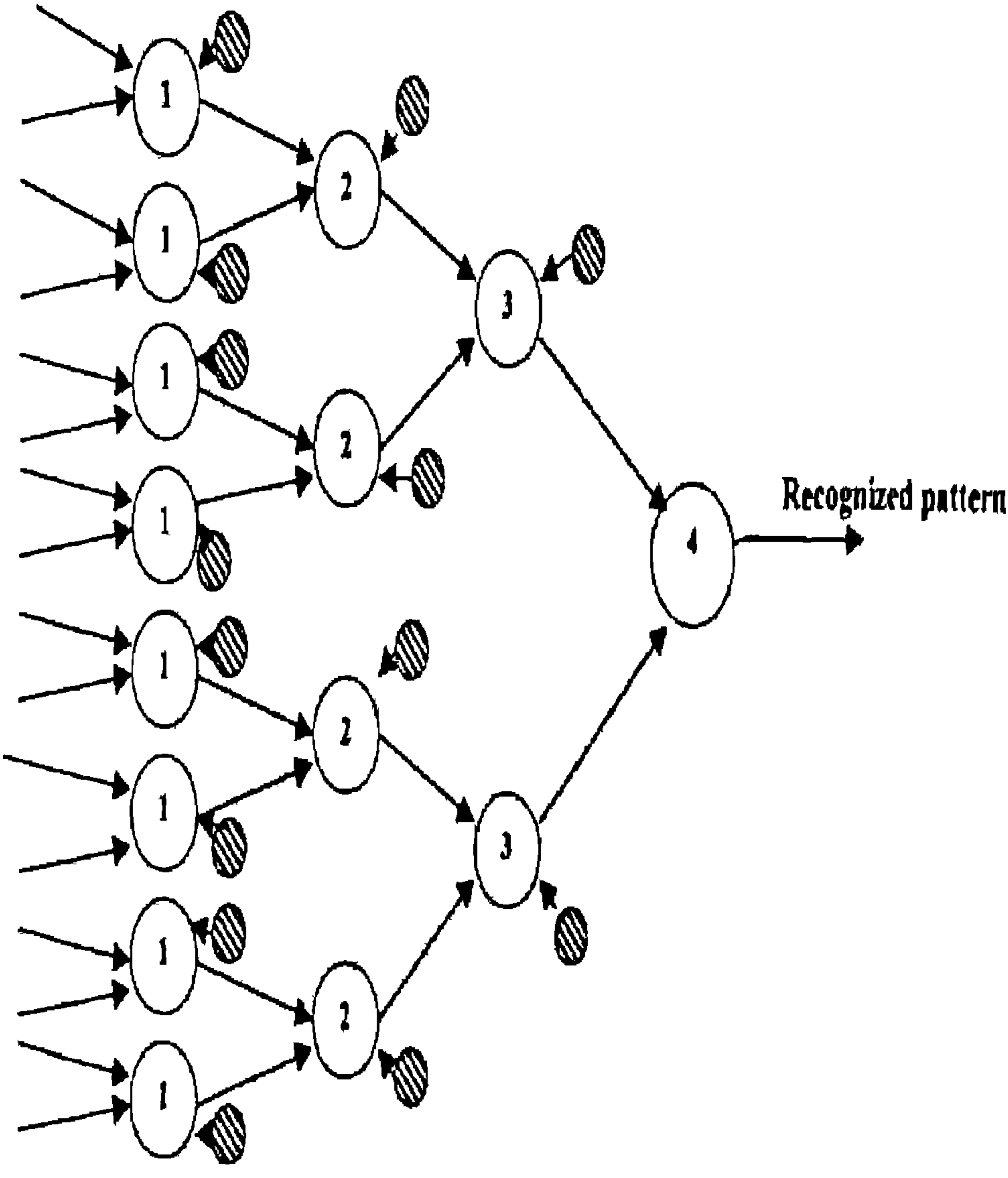


Fig 1 Multiple Neural Network model

2.3.4 Working of MNN

The set of exemplars is given to the Neural Network in the form of binary strings The Neural Network groups these patterns into pairs of two Every network in the first layer L_1 takes one pair as exemplars The test pattern is also presented as a binary string, to all of these networks Each of these networks then compares it with both the exemplars for that network and outputs the best match for that test pattern. The best matching exemplars obtained from the networks in layer L_1 are fed as training patterns to the next layer of networks i e , L_2 The number of networks in this layer is equal to half the number of networks in the layer L_1 The test pattern is fed to all these networks again The process is continued till the number of networks in the layer becomes one This is the last layer and the net work in this layer outputs the final recognized character

When both upper case and lower case letters of English Alphabet are taken as the reference data, the number of networks in the outermost layer, that is L_1 is 26 and the number of layers is 6 When only either upper case or lower case letters are taken, the number of networks in layer L_1 is 13 and the number of layers is five

A schematic diagram of the MNN used in the present work is shown in fig 1

Chapter 3

DESIGN AND IMPLEMENTATION

3.1 Overview

The aim of Optical Character Recognition (OCR) is the recognition of individual characters by the computer when these are presented to the recognition system in the digitized form Currently, a lot of published material exists on OCR of many different languages including English, Chinese, Japanese, Arabic and Indian languages like Devanagiri, Telugu, Tamil, etc Several surveys on OCR of printed and handwritten characters can also be found in the literature Some of these have been discussed in the previous chapters A number of commercial OCR systems are also available for some of these languages

A conventional OCR system consists of a series of steps, which may be represented by a block diagram as shown in fig 2 The digitized or scanned form of the character image is obtained from the page containing it and is a matrix of intensities of different individual pixels (picture elements) of the character The preprocessing stage consists of two steps The first one involves the conversion of multiple gray-levels (image intensities) to two levels (binary image) which is called thresholding This technique facilitates the identification of the pixels that contribute to the description of the shape of the character Unlike image processing techniques where gray-level images provide significant information at each location in terms of intensity of a pixel, the character recognition task requires only a sort of shape description and thus a binary image would suffice for this kind of study It is to be noted that computer handling of binary image is much more space and time efficient than a gray-level image The second step involves reducing the original thickness of

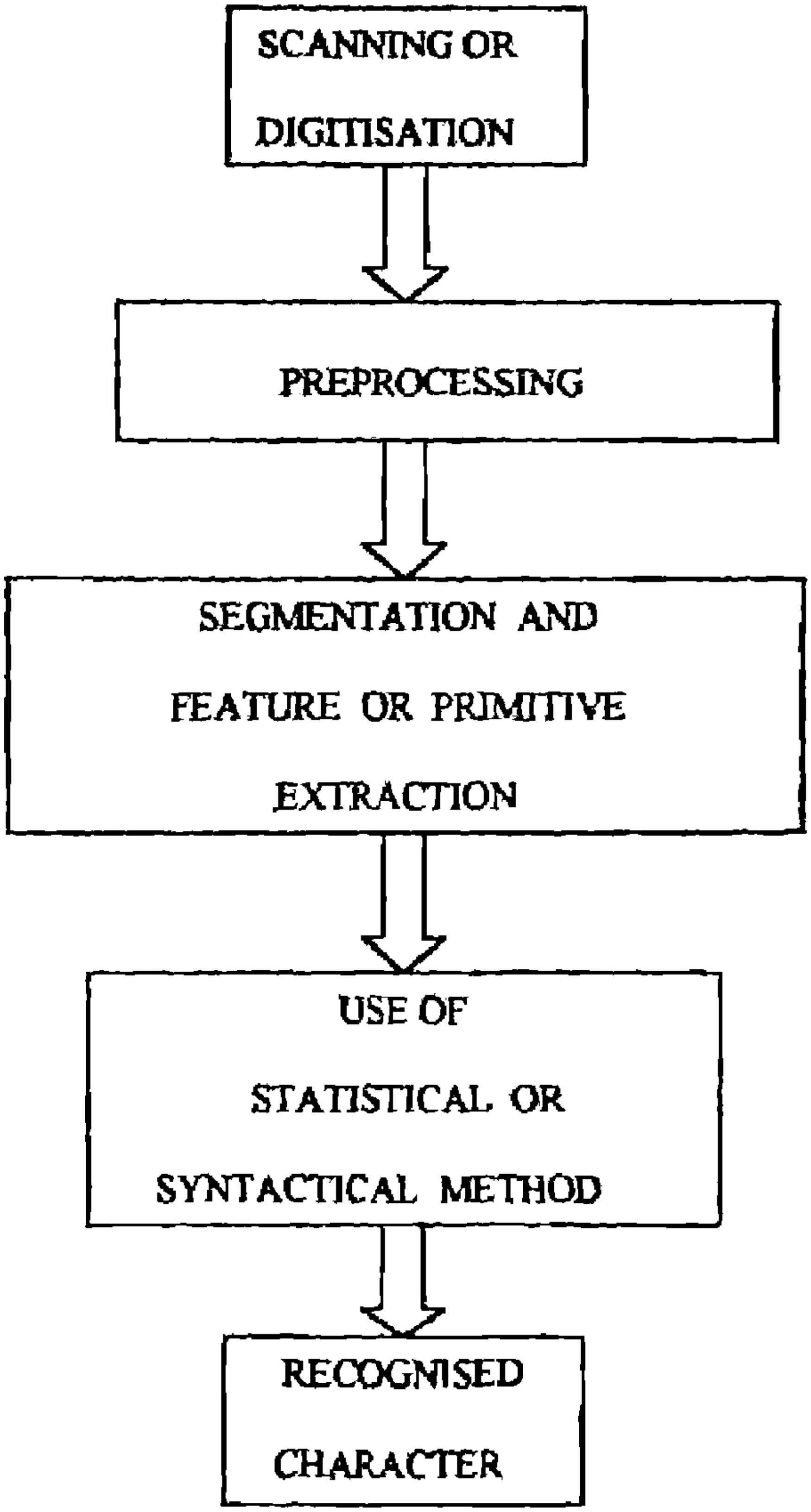


Figure 2 A generalized block diagram of conventional optical character recognition

the characters to a value which is convenient for recognition (called skeletonization or thinning)

The approaches to character recognition can be broadly classified into two methods

a) *Feature Mapped Recognition* In this method, recognition is achieved by identifying primitives/features in the digital image consisting of the character The preprocessed image is first segmented or separated into different meaningful elements, called features or primitives These primitives are then recognized individually and thus the character is recognized by searching for the occurrence of one or more of these primitives or features Statistical decision functions may be computed during the search process Syntactic or structural pattern recognition is also an example of feature mapped recognition because the pattern to be recognized is expressed as a string of symbols and these symbols are obtained by coding the features which are first extracted Rules of syntax to form different legal strings of the pattern grammar are first decided and the test pattern, also expressed as a string, is verified as to whether it is legal within the grammar

b) *Image Mapped Recognition* Here, recognition of the complete or part of the character image is achieved The whole image, rather than different primitives extracted from it, is presented to the recognizing mechanism Artificial Neural Network based recognition belongs to this category of recognition because the whole image is presented to the Artificial Neural Network for recognition In the present work the approach of image-mapped recognition was adopted and the block diagram of the proposed character recognition scheme is shown in fig 3

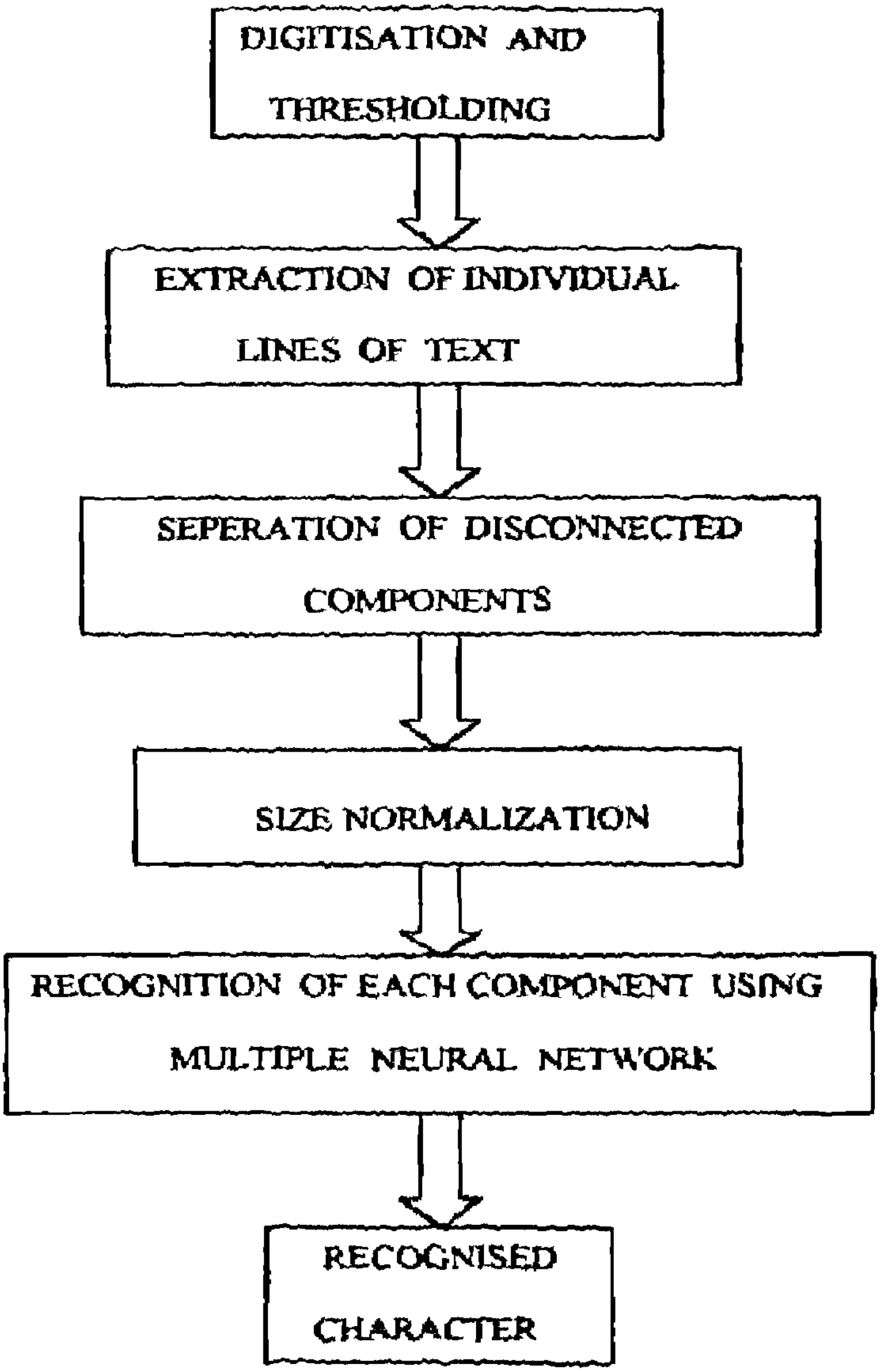


Figure 3 Block diagram of the proposed optical character recognition system

3.2 Image Acquisition

The OCR system takes the image file as an input (a *.bmp* windows format file) which passes through various stages of preprocessing The initial processing stage in any OCR system requires conversion of paper-based image or text to a digitized image The digitization can be achieved using a HP Deskjet scanner Binarization converts the document into discrete array of numbers, which can be stored and processed The header of the image file gives information such as image height and image width The length and content of the header vary with the format of the image file

The process of loading the bitmap image from the disk to the memory is performed by the function **loadBMP( )**. The **loadBMP( )** function is passed a **Cfile** object identifying the bitmap file which will already be opened The first step is to read in the bitmap header This is stored in a type BITMAPFILEHEADER structure, and provides a variety of information about the bitmap including the file size and the location in the file of the data bits This header structure has a member variable **bfType** The code checks to see that this variable contains "BM", indicating a bitmap file The code then reads the header into a type BITMAPINFOHEADER structure The next step is to read the color table data Finally, the bitmap bits, the data that actually defines the image are read The variables pbmpi and pbits are set to point at the data This information is used to create a **hbmp** object by calling the windows API function **CreateDIBitmap( )**

The document when digitized is stored as bits in the image file An image pixel is stored as zero and background pixel is stored as one Since only bytes can be accessed from the memory, we get one byte from the file in one access Decoding this byte into eight individual pixels is known as unpacking Getting the **hbmp object** obviates the need for unpacking The **GetPixel( )** function can be used top directly obtain the color values of individual pixels

3.3 Displaying The Document

The display function draws the specified bitmap onto the device context, which is passed to the function as an argument. This function displays the bitmap by calling the windows API function **StretchBlt( )**. This function uses the operation SRCCOPY, which copies the bitmap without alteration to the destination. The **StretchBlt( )** function has the ability to stretch or compress the bitmap This is done if the destination width and height differ from the source width and height For the present purpose, the destination width and height are fixed so that the bitmap is always displayed at the specified position and with the specified standard size on the screen

3.4 Preprocessing

Preprocessing is that stage of recognition, wherein the raw image obtained by scanning a picture is converted to a form acceptable to the recognizer To extract individual characters from the digital image in a form ready to be presented to the neural network for recognition, preprocessing was performed as described I the following sections For character recognition, this involves conversion of a gray scale image to a binary image In our present work this is carried out while scanning, by scanning the document as a bi-level image Further steps involved are the isolation of individual characters, and aligning them both horizontally & vertically with respect to the central axes

3.4.1 Extraction Of Individual Lines Of Text

To extract individual characters from the document, lines of text have to be isolated The function **linesep( )** is used for this purpose The number of black pixels in scanline of the page is counted A line starts at a scanline with non-zero number of black pixels and it ends at scanline with no black pixels The beginning and the height of all lines in the page are then obtained and the number of lines in the page is counted The lines with height less than five pixels are ignored to account for noise or any random pixels The remaining lines are stored in an array of two-dimensional matrices called **line.** Function **linesep( )** is used to perform these operations Ithis funtion is described below

<u>**Function linesep( )**</u>

Variables used.

Count array containing the count of black pixels for each scans line

Lineht - array containing the height of the lines in pixels

Linebgn array containing the beginning scanline number of the line in the bits matrix

Lineno counter used to count the number of lines

The algorithm used for the process is given below

```
for each scanline
{
        count the number of black pixels from the bits matrix
        store them in count array
}
for each scanline
{
        if count is not zero and flag is not set
        {
```

```
            increment line count
            save the scanline number as the line beginning
            set flag
      }
if count is not zero
      increment lineht for that line
if count is zero and flag is set
      reset flag
}
if lineht < 5 ignore the line
else allocate the memory for the matrix to store the line
store the line in the array line
```

Thus, the separation between lines is obtained and the lines of text are extracted

3.4.2 Isolation Of Single Characters

Individual characters are extracted from the lines by obtaining the separation between the letters The algorithm used for this is similar to the one used to separate lines of text The isolation of characters is performed by the **lettersep( )** function This function takes the line number as an argument and extracts the letters from the specified line.

In order to be able to group the recognized letters subsequently into words and lines, it is necessary to keep a count of number of word sin each line and the number of letters in each of these words This task is also performed by the **lettersep ( )** function

The **lettersep( )** function takes as argument the line number of the line to be processed The line number is used to index the specified line in the global **line** array

A count of the number of black pixels in each column of the line matrix is maintained This count is used to find the separation between letters The count of black pixels for each column of the line is examined, starting from the first column The first column with non-zero count is considered as the beginning of the first character of the line The letter ends at the next column with zero count of black pixels The second letter starts at the next column with non-zero count of pixels and ends wit the following column of zero count This process is continued till the last column of the line is reached The count of number of letters is maintained and the minimum blank space between two successive letters is determined If the blank space between any two letters is more than twice the minimum blank space then it is considered as a word separator The count of number of words in the line is maintained

The letters extracted from the line are stored in an array of matrices called **letters**

Function lettersep( )

Variables used

count Array containing the number of black pixels in each column of the line matrix

blank Number of consecutive blank columns between previous letter and present letter

minBlank Minimum number of consecutive columns with zero black pixels

ltwidth Array containing the width of all the letters of a line

ltcount Number of letters in the line

nword number of words in the line

nletterw array containing the number of letters in each word of the line

The algorithm is decribed in the following pages

```
for each column of line matrix
{
        count the number of black pixels
        store them in count  array
}
For each column of line matrix
{
        if count is not zero  and flag is not set
        {
                set flag
                increment letter count
                update minblank
        }
        if count is not zero  increment ltwidth
        if count is zero  and flag is setreset flag
}
//calculating the number of words
```

```
For each column of line matrix
{
        if count is not zero  and flag
        {
                set flag
                if blank>minblank..
                {
                        increment wo
                        set wflag
                }
```

is not set

ord count

```
            if count is zero and  flag is set
            {
            reset flag
            if blank>minblank and wflag is set  then reset wflag
            }
      }
}
//calculating number of letters in each word
for each column of line matrix
{
      if count is not zero and flag is not set
      {
            set flag
            if blank>minblank
            {
                  increment nword
                  set wflag
            }
            if wflag is set
                  increment nletters
      }
      if count is zero and flag is set
      {
            reset flag
            if wflag is set and blank>minblank
      reset wflag
      }
}
```

allocate memory for all the letters in the line

store the letters in the letter array

Thus we obtain characters of the document in the form of binary matrices with varying widths and heights.

3.4.3 Centering And Size Normalization

The individual characters obtained are of non-uniform size and are not centrally aligned. The size normalization is performed simultaneously with centering the letter on to a binary matrix. The Centering is done in two stages. In the first stage, the letter is aligned vertically i.e., with respect to the central horizontal axis and the height is made uniform. The average height of all the letters is taken as the standard height. This is done by the function **VAlign( )**.

The **VAlign( )** function first computes the height of the actual letter by ignoring all the horizontal rows with zero black pixels. This letter is then centrally placed in a matrix whose height is equal to the standard height and whose width is equal to the width of the actual letter. Thus we obtain letters of equal heights and varying widths.

To make the width of the letters uniform, the letters are stored centrally in a matrix of standard size. This is done by the function **HAlign( )**. The matrix on to which the letters are centered is called the Isolated Character Matrix or ICM. If the width of the letter is more than the width of ICM, the letter is truncated equally on both sides, if the width of a letter is less than that of the ICM, the letter is padded equally on both sides with columns of white pixels.

The functions **VAlign( )** and **HAlign( )** are described in the following pages.

Function VAlign( ):

For all the letters in page

```
{
        find the beginning scanline of the letter
        find the ending scanline of the letter
        letterht = letbgn-letend
        store the letter vertically centered in a matrix of uniform height
}
```

Function HAlign( ):

For all the letters in page

```
{
        if width of letter > standard width
                truncate the letter equally on both the sides
        if width of letter < standard width
                pad the letter uniformly on both the sides
        store the letter centrally in the ICM
}
```

After processing the letters through HAlign( ) and Valign( ), we obtain letters that are centered and are of uniform size

3.4.4 Details Of Preprocessed Information

The document information obtained from preprocessing is stored in a structure variable called PI of type PageInfo The structure PageInfo is declared as

```
struct PageInfo{
      int        Width;
      int        Height;
      int        NoOfLines;
      int*       NoOfWordsInLine;
      int**      NoOfLettersInWord;
      int        NoOfLettersInPage,
      matrx*     Letter,
};
where
```

Width is the width of the page image in pixels

Height is the height of the page image in pixels

NoOfLines is the number of lines of text in the page

NoOfWordsInLine is an integer array that contains the number of words in each line of the page

NoOfLettersInWord is a two-dimensional array that contains the number of letters in each word for all the lines in the page

NoOfLettersInPage is the total number of letters in the page

Letter is an array of matrices in which each matrix contains the ICM for a single letter in the page

Memory is allocated appropriately to each of these variables and the relevant information is stored in them. The information in this structure is used for all the further processing

3.5 Experimentation

Earlier work to solve the problem of character recognition involved conventional PR techniques like primitive/feature extraction code generation of primitives and template matching Such algorithmic methods are not immune to noise in the character image Also, if characters are not properly thinned, the recognition mechanism fails to perform with full efficiency A Neural Network approach to the problem could overcome these drawbacks owing to the robustness of artificial Neural Networks to noise and distortion

Several experiments have been carried out using several strategies and the strategy that gave satisfactory recognition was adopted Details of the experiments conducted are given below

3 5.1 Feeding The Complete ICM

Initially the ICM was fed as a whole to the Neural Network Due to limitations in the capacity of the Neural Network, the ICM was scaled and the bits of the ICM were as a single string The ICM was scaled such that the length of the string formed by concatenating the consecutive rows is less than the capacity of the Neural Network The scaling algorithm used initially is given as follows

The Neural Network dimensions are fixed to a size of 8 rows by 6 columns (i e , 48 neurons) A simple scaling transformation is used to reduce the isolated character dimensions to the standard dimensions The ICM is divided into scale windows (SW) whose dimensions are same as that of the standard dimensions Now, all or any one of the scaled windows can be presented to the Neural Network The number of scaled windows of dimension 8x6 that can be obtained from an ICM of dimension h x w is

$$\text{int}\left(\left(h \times w\right)/48\right)$$

A pixel from an ICM is mapped on to the *i*th scale window if the remainder of pixel position in ICM divided by number of windows is *i*, i e., pixel x is mapped onto scaled window *i* if (position of x in ICM) mod (number of windows) is equal to *i*, where operation x mod y gives the remainder of x divided by y

In such a transformation, it is found that with dimension reduction there is multiplication in the number of characters generated when the starting pixel in the ICM is different And these extra characters may be input for testing, sometimes the skeleton of the character is obtained with this transformation

It has been observed that with this scaling algorithm, the noise and distortion in the letters has increased and hence the recognition rate was very low Therefore another algorithm was used for scaling. With this algorithm the ICMs had a better recognition rate since the noise and distortion was reduced This algorithm is as follows

Column ratio is the ratio of number of columns in the ICM and the number of columns in the standard ICM Row ratio is the ratio of number of rows in the ICM and the number of rows in the standard ICM Initially, the ICM is divided into column ratio number of scaled windows the width of which is that of standard ICM and the height of which is same as that of the actual Icm All the rows of the ICM are then mapped onto these scaled windows The mapping is done such that the rows beginning at the *i*th column are mapped to the *i*th scale window.

Now, each of the scale windows is further divided into row ratio number of scale windows, the width and height of which are that of the standard dimensions i e , the scale windows have a width of 6 and a height of 8 All the columns of ICM are the mapped onto these scale windows The mapping is done such that the columns beginning at the *n*th row are mapped onto the *n*th scale window

The noise and distortion of the letters was reduced as compared to the letters obtained by using the first scaling algorithm But the recognition rate was not satisfactory and the time for the recognition was very high Since the time taken for recognition increases exponentially with increase in thew length of input patterns, it is impracticable to feed the whole ICM to the Neural Network Thus other strategies were experimented with, wherein the Neural Network was fed width column or rows of individual characters thus greatly reducing the time taken for recognition

3.5.2 Hashing For Size Normalization

The ICMs of all the letters from a document do not have a uniform width and height. In order to make the height of the ICMs uniform, the average line height was calculated and all the lines were reduced to the average height This was done by superimposing the first two rows in the line matrix for lines of height greater than average height Generally, the line height variation is one or two rows Since the line height is made uniform, the letters extracted from the lines will be of uniform height The letter width is made uniform by using a hashing function. Three different hashing functions were experimented with The hashing functions used are briefly described below

1 Each horizontal row of bits in the ICM, which is a binary string, is converted to a decimal number To get a binary string of length eight, the decimal number was divided by (2^8-1), the remainder of which is converted back to a binary string All the rows of the ICM are hashed to rows of uniform length eight and thus the letters obtained are of uniform width and uniform height

2 Another method, which exaggerates variability throughout the string of digits comprising the key, is called *radix transformation* Briefly, each digit of the key is evaluated as if with respect to the base (base+1), i e , a base one higher than the normal base It is then reinterpreted with respect to the original base We have been expressing our rows of binary strings from ICM as integers to the base ten, we evaluate the number as if the digits were in a base eleven number system After the radix transformation, extraction was used to ensure that the number would be of standard length The last eight digits in each row of the ICM are extracted Thus all the letters obtained are of uniform width and uniform height

3. In the third method the last eight digits in each row of the ICM are extracted Thus all the letters obtained are of uniform width and uniform height

The letters, after being hashed to a uniform width are fed to the Neural Network column-wise and row-wise Several experiments were conducted with varying widths from 5 to 30 bits And most satisfactory results were obtained with a width of eight

3.5.3 Superimposition Techniques

Superimposing is another technique that has been used to make the size of ICMs uniform Superimposing avoids the loss of information when the size is reduced It can also be used to reduce the effect of shifting of letters if they are not centrally located within the ICM The Superimposing of columns was done, to make [illegible]

augment

In the first technique the letter is centrally arranged with respect to the central vertical axis Those columns of the letter which fall outside the ICM are superimposed with one another All the columns on the left of the letter are superimposed with one another to get the first column of the ICM and the columns to the right of the letter are superimposed to get the last column

In the second technique, which tries to minimize the effects of shifting, each column is superimposed on the following column This accounts for any shifting of the character matrix side-ways The columns obtained after superimposition are fed to the Neural Network

3 6 Recognition

A technique that gave more satisfactory results than the above mentioned methods was finally implemented for the recognition mechanism The following sections describe the procedure used for this technique

3.6.1 Preparation Of Data

The Neural Network recognizes binary bit patterns Both the exemplars i e , the standard patterns and the test patterns are to be presented as strings of zeroes and ones The Neural Network takes a file name (fname) as argument The exemplars are presented in a file with name fname with *bit* extension The test patterns are presented in a file of same name with *.mod* extension The patterns are to be separated with newline characters The Neural Network outputs the recognized patterns in file with the same name as fname with *res* extension The string number of recognized patterns is stored in a matrix called **reco**

If the ICM is presented as a whole to the Neural Network, The length of the pattern will be equal to width times the height of the matrix As explained above, the capacity of the Neural Network is restricted to 50 bits per string Hence the ICM has to be scaled to a size that gives a length of string less than this size Also, the time take for recognition increases exponentially with the length of the pattern The amount of recognition time for this approach makes it impracticable Moreover, the scaling of the letter introduces distortion and might increase noise in the character image This reduces the possibility of correct recognition

In view of all the above factors, the ICM is fed to the Neural Network column-wise Thus the first column of the ICM of the test letter is compared with first columns of all the standard letters Similarly all the other columns of the test patterns are compared with the corresponding columns of the standard patterns Since the first columns of some standard letters may be same, the same pattern might appear as the exemplar more than once This is redundant So only the distinct columns are selected and presented as exemplars Each of these exemplars then represents one or more of the standard letters The information about the letters represented by each of these exemplars is stored in an array called **info**

So to prepare the data for recognition it is required to first write individual columns of both standard and test letters into separate files Further the standard patterns have to be processed further to get only the distinct strings

The Neural Network is then called for each column and the corresponding files for the test and the standard patterns are fed to it as the input The Neural Network outputs the result in a matrix called **reco**. **reco** is a two dimensional array the first dimension of which represents the letter number and the second, the column of that letter An element (*i* , *j*) of the **reco** matrix contains the string number of the nearest match for the *j*th column of the *i*th letter in the document

The information stored in **reco** is used to identify the best-matched letter Two functions **StoreInFiles( )** and **GetDistinct( )** are used to prepare the data for the Neural Network These functions are briefly described below

Function **StoreInFiles( )**

This function takes the ICMs that are centered and are of uniform size For each column of the letters, a file with *.mod* extension is opened and the corresponding column of all the letters is written into the file For example, the first columns of all the letters are written into a file called *test0.mod* and so on So the number of files is equal to the number of columns in the ICM and number of patterns in each file is equal to the number of letters in the page The function is as follows

```
for each column
{
        for each letter in page
                {
                        read the bits of the column
                        write them in corresponding .mod file
                }
}
```

Function **GetDistinct( )**

This function is used to prepare the standard data i e, the exemplars for the Neural Network Each file is read and all the distinct strings are obtained and are written into a file with the same name and *.bit* extension The letters represented by each row are stored in a matrix called **info**

This function is explained below

```
for each file of standard patterns
{
        read all the patterns into an array
        for each pattern in the array
                {
                        if the pattern has not been written already
                              write the pattern into the corresponding  bit file
                        write the information into the info matrix
                }
}
```

3.6.2 Presentation Of Test Patterns For Recognition

Preprocessed test patterns that are obtained by the above described methods are now presented to the Neural Network Since the character patterns are presented column-wise, the Neural Network is called once for each column of the ICM The Neural Network, as described above, outputs the string number of the nearest match in **reco** array Thus, we get a set of potentially matching letters for each column for all the letters

3.6.3 Identifying The Best Match

From the set of all possible matches, the letter that is the best match for a given character has to be identified This process needs the information stored in **reco** and **info** matrices

For **each** letter of the document the **reco** matrix contains the string number of the best-matching exemplar for each column of the ICM This string number is used to index into the **info** matrix to find out all the standard letters that are represented by the exemplar corresponding to that string number Thus we obtain all the potential matches for a letter The letter that occurs maximum number of times in this set of potential matches is the nearest match for the letter

In order to find the best match, a count is maintained for each of the standard letters and the count is incremented by one, for each occurrence of the standard letter in the set of potential matches The standard letter with the maximum count is the best match and is stored as the recognized letter In some cases, there is more than one letter, which has the same maximum count Then all these letters are stored in a matrix called result

3 6.4 Categorizing the Recognized Letters

The recognized letters are classified into three groups based on the way in which the Neural Network recognizes them This is done to facilitate the analysis of the performance of the system A function called **group( )** is used for this purpose This function is described below

<u>Function **Group( )**</u>

This function categorizes the letters into four groups The first group contains letters, which are uniquely, and correctly recognized i e , the Neural Network outputs a single recognized letter The second group contains letters for which the Neural Network outputs several recognized letters out of which one is correct The third group contains incorrectly and uniquely recognized letters The fourth group contains letters for which the Neural Network outputs several recognized letters, all of which are incorrect An array containing the characters (text) from page is used to compare the recognized letters output by Neural Network to determine whether the letters are correctly recognized The percentage of correctly recognized letters is also calculated All the above information is stored in a file *res dat* This information is used during testing to evaluate the performance of the Neural Network

3.6.5 Forming The Document

The recognized letters from the result matrix are grouped into lines and words A function named **Assemble( )** is used for this purpose The information about the number of lines in the page, the number of words in each line and the number of letters in each of these words is stored in a structure **PageInfo** as described earlier The function **Assemble( )** makes use of this information to assemble the recognized letters into a document. This document is output as a text file

TESTING AND RESULTS

4 1 Testing

The system was tested with scanned images of various documents Documents with different fonts and character sizes were used The standard letters of all these fonts are also printed and scanned These standard letters are used as the reference set of letters for that font The results presented here are for the font **Arial** with a character size of **12**

The pages are scanned using HP Deskjet 4c scanner with a resolution of 75 dpi The scanned images are stored as *.bmp* files The document containing the standard letters of the font is processed first to get the reference data This is then used to compare and recognize the document containing the characters of that font
For testing, the documents containing a only a single case (upper or lower case) alphabet have been considered to make the experimentation simpler All the three methods in chapter 3 have been used to test the data The results indicate that the third method gave best results for both upper case and lower case letters The characters are divided into groups depending on how they are recognized This grouping shows that the ambiguity within recognized letters was slightly lesser when the hashing was used for size normalization

The performance of hashing and superimposition techniques was relatively poorer in case of lower case letters The third technique used gave consistent results for both upper and lower case letters

4.2 Results

The results of evaluation tests are presented in the following pages The bitmap image of the document used in testing is presented along with the document as recognized by the present OCR system

The recognition percentage with various techniques is also shown in the form of graph The table below the graph shows the performance analysis of the techniques used The experiments conducted are briefly described below

Experiment 1 .

In this experiment hashing was used to make the width of the letters uniform The hash function (Hash function-1) used was described in chapter 3

Experiment 2

This method makes use of the superimposition techniques as described in chapter 3

Experiment 3 .

This method was finally adopted for the recognition mechanism of the present system In this method ICM has been fed column-wise to the network The method is described in chapter 3

Groups:

Group 1 The first group contains letters, which are uniquely, and correctly recognized i e , the Neural Network outputs a single recognized letter

Group 2 The second group contains letters for which the Neural Network outputs several recognized letters out of which one is correct

Group 3 The third group contains incorrectly and uniquely recognized letters

Group 4 The fourth group contains letters for which the Neural Network outputs several recognized letters, all of which are incorrect

AT THE TIME WHEN I STOOD IN THE CHURCHYARD READING THE FAMILY TOMBSTONES I HAD JUST ENOUGH LEARNING TO BE ABLE TO SPELL THEM OUT MY CONSTRUCTION EVEN OF THEIR SIMPLE MEANING WAS NOT VERY CORRECT FOR I READ WIFE OF THE ABOVE AS A COMPLIMENTARY REFERENCE TO MY FATHERS EXALTATION TO A BETTER WORLD AND IF ANY ONE OF MY DECEASED RELATIONS HAD BEEN REFERRED TO AS BELOW I HAVE NO DOUBT I SHOULD HAVE FORMED WORST OPINIONS OF THAT MEMBER OF THE FAMILY NEITHER WERE MY NOTIONS OF THEOLOGICAL POSITIONS TO WHICH MY CATECHISM BOUND ME AT ALL ACCURATE FOR I HAVE A LIVELY REMEMBRANCE THAT I SUPPOSED MY DECLARATION THAT I WAS TO WALK IN THE SAME ALL THE DAYS OF MY LIFE LAID ME UNDER AN OBLIGATION ALWAYS TO GO THROUGH THE VILLAGE FROM OUR HOUSE IN ONE PARTICULAR DIRECTION AND NEVER TO VARY IT BY TURNING DOWN BY THE WHEEL WRIGHTS OR UP BY THE MILL

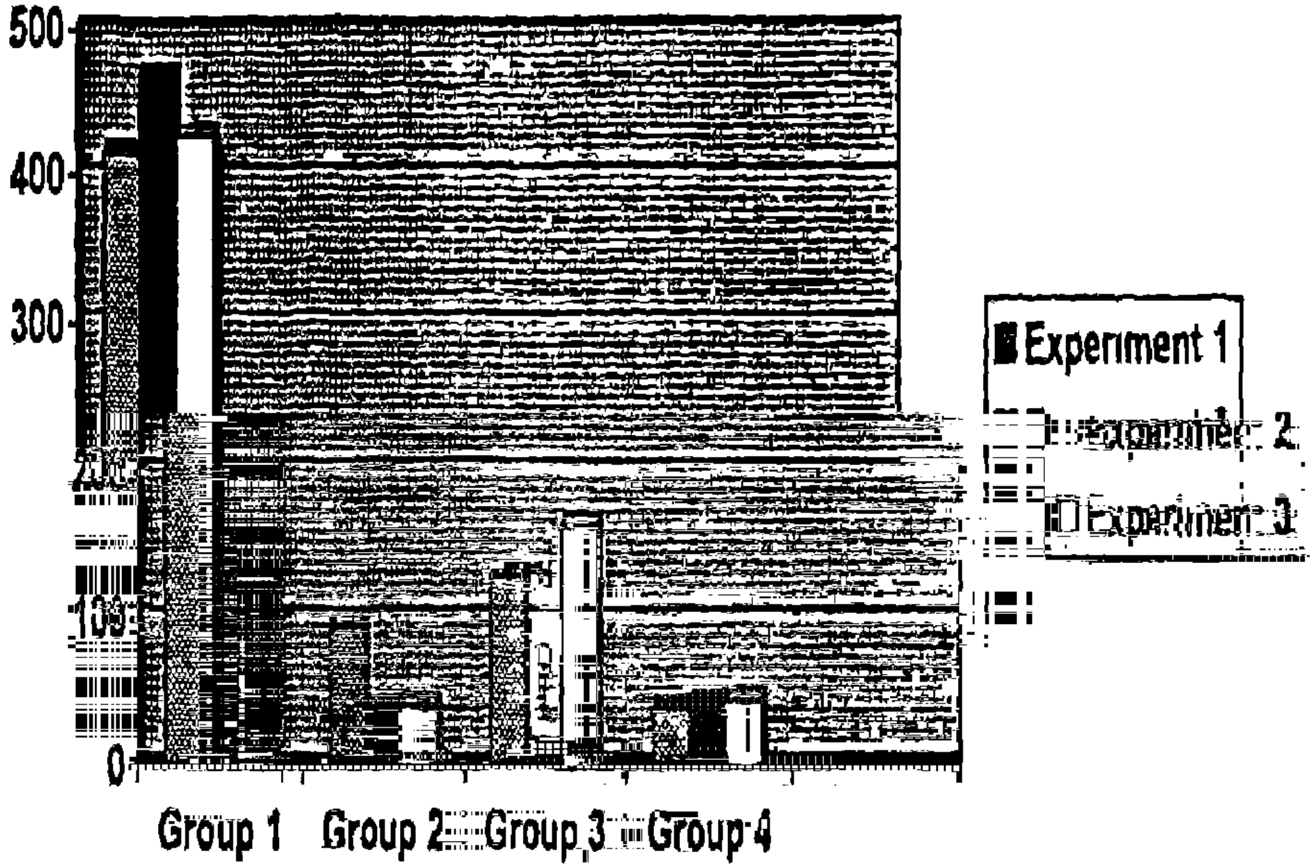
Cap 1

AT THR TIMB WHBK I BTQQD IB THB QHLRQHYARQ
RBADING
THB FAMILY TOMBBTOFBB I HAD JUBF ENGUCH
LSARKILE
FO JB AJLS TO FFBLL TKSM DUT MY
QQKBTRLQTIQK BVBN
QF THEIR BIMPLB MBABINQ WAB NQT LERY CDFRECT
FOR I
FEAE WIFE OF THE AEGVE AS A COMPJIMEKTARY
RBFBRBMQB TO MY FWTHBRB BAALTATIQK TQ A
BBTTBR
WQRLQ ANQ IF ANY QNE QF MN QBQBABBD
RBLATIGNE HAD
BEEP RLFERFED FO AF SKLCA I YAVB KO QDLST
I
FYOLLQ MAMB FORMBQ WDJET QPIKIQNB QF THAT
MBMBBR QF THB FAMILN MBITHER WBRB MY NQTIGNE
GF
THEDLDGIOAL POSITIGNS TG WHIOH MY DAFECIIFM
SQUMQ MB AT ALL AQQLRATB FQR I HAVB A
LIVBLY
RBMBMBRANQB THAT I BUFPQSEQ MY DBQLARATIOM
THAF I

[illegible]
TIH
VILLAGE [illegible] OUR [illegible]
DIRRQTIQK [illegible] WNE [illegible] TO [illegible] VARY [illegible] BY [illegible] TURNI
DOWN BY
THB WHBBL WRIGHTS [illegible] THE [illegible]

Recognized Cap1

Cap1



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	416	94	129	41	75 00
Experiment 2	469	41	126	44	75 00
Experiment 3	428	42	164	46	69 12

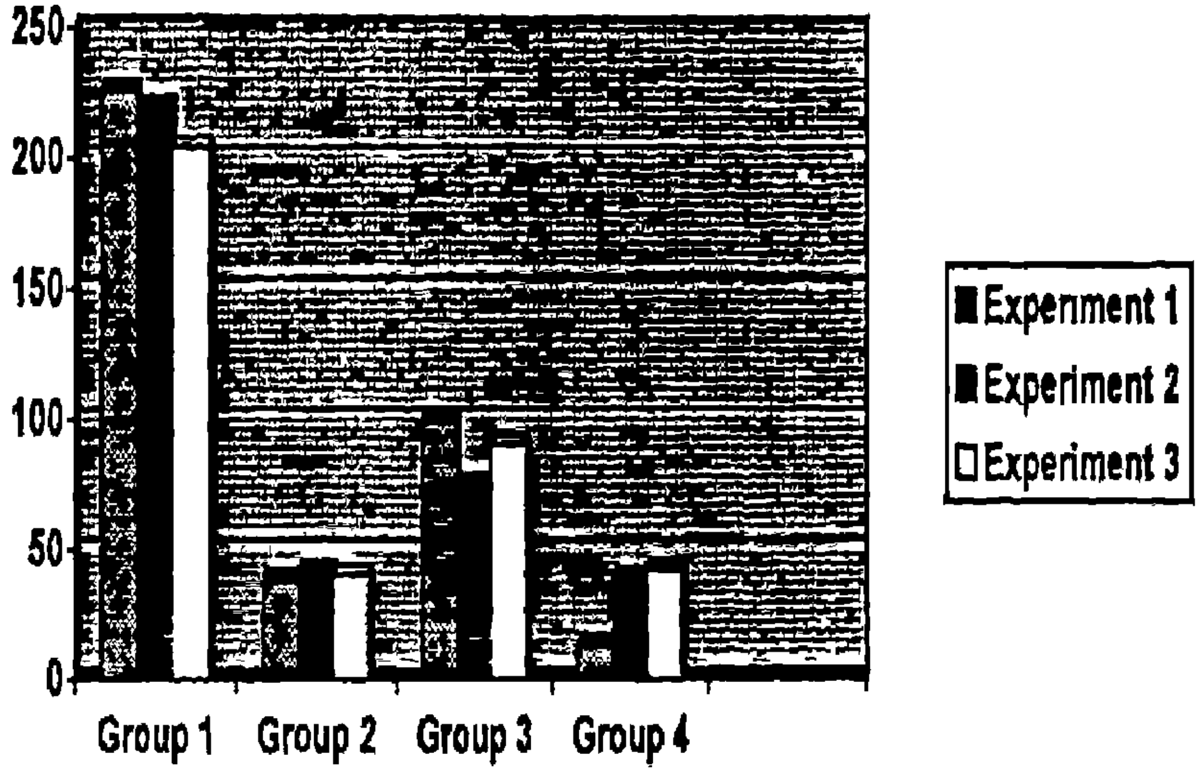
THE SOUND OF HER IRON SHOES UPON THE HARD ROAD WAS QUITE MUSICAL AS SHE CAME ALONG AT A MUCH BRISKER TROT THAN USUAL WE GOT A CHAIR OUT READY FOR MRS JOES ALIGHTING AND STIRRED UP THE FIRE THAT THEY MIGHT SEE A BRIGHT WINDOW AND TOOK A FINAL SURVEY OF THE KITCHEN THAT NOTHING MIGHT BE OUT OF PLACE WHEN WE HAD COMPLETED THESE PREPARATIONS THEY DROVE UP WRAPPED TO THE EYES MRS JOE WAS SOON LANDED AND UNCLE PUMBLECHOOK WAS SOON DOWN TO COVERING THE MARE WITH A CLOTH

Cap 2

THB BQLLK QF HBR IRON SHOEB UFDN TKB KARQ
ROAD
AAB QUITE MUSICAL AF FKB QWMB ALQNQ AT A
MCCH
SRIEKSR TRQT THAN UBUAL WB GQT A OHAIR OLT
RBAQY
FQR MRB JOEB ALIGHTING ANQ FTIRRBD UP THB
FIRE THAT
THEY MIGIT FBS A SRIGHT WIBDQWANQ TQOX A
FINXL
BURVBN QF THB KITQHBN THAF NGTIING MIGKT BB
QUT QF
PLAQB WHBN WB HAD QQMPJEFLQ THSFB
PRSFARATIQMB
THEY QJOVB UP WRAPRBQ TQ THB BYBS MRB JGE
WAF
EQQM LANDBQ ANQ UMQLB PUMELECHGGKWAF BQQN
DDWN TG OGVSRIKG TKS MARB WITH A QLQTH

Recognized Cap2

Cap 2



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	226	38	99	13	70 21
Experiment 2	220	42	75	39	69 68
Experiment 3	204	40	90	42	64 89

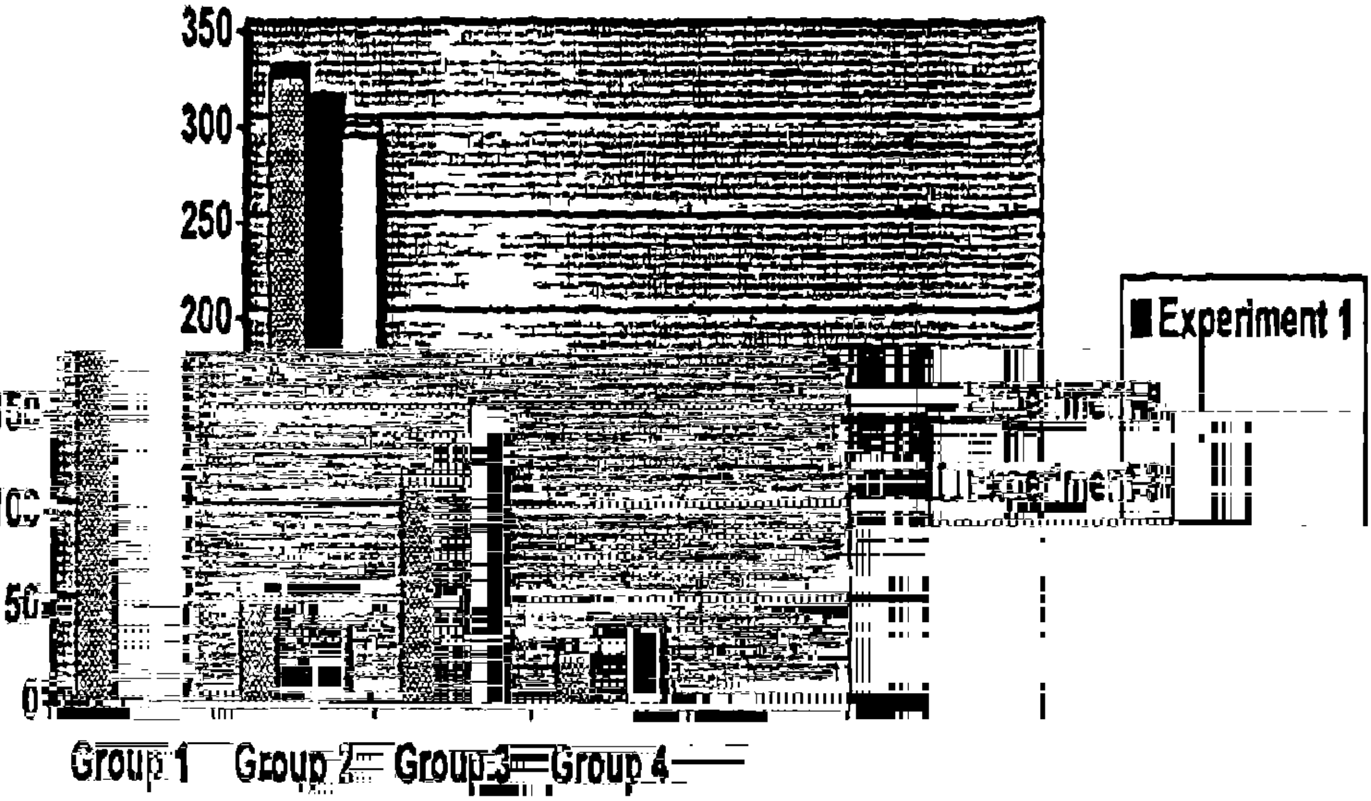
THE BOOK IS APART FROM ALL THE OTHER DICKENS NOVELS IN THAT IT HAS A THEME WHICH PREVENTS IT FROM BEING PICARESQUE LIKE NICHOLAS NICKLEBY AND FORBIDS IT TO HAVE ANY SUCH EPISODE AS MARTIN CHUZZLEWIT HAD OR TO HAVE NO OTHER UNITY THAN THAT OF ITS CENTRAL CHARACTER LIKE DAVID COPPERFIELD THE THEME IS PIPS DISCONTENT WITH HIS ACTUAL BENEFACTORS FIRST WITH JOE LATER WITH PROVIS HIS INNOCENT MISATTRIBUTION OF HIS FORTUNE TO MISS HAVISHAM HIS SUPPOSITION THAT SHE MUST INTEND ESTELLA FOR HIM BUT THIS THEME CARRIED OUT BY DESIGN PIPS DISCONTENT HIS UNWILLING INJUSTICE TO JOE IS CONFRONTED WITH WEMMICKS PERFECT SERVICE OF HIS AGED PARENT

Cap 3

TKS SQQK IS AFART FRQM ALL THB QTHER DICXEKE
NQVBLS IN THAT IT HAB A THEME WHIOH PRSVBKTS IT
FRGM EEIPG FIOARBEQLB LIKB NIQHQLAB BIQXLBBY AND
FQRBIKB IT TQ HAMB ANN SUQH BPISDDE XE MARTIK
QHUZZLBWIT HAQ OR TO HAVE KO OTKBR LNITY THAN
TKAT QR ITB QBNTRAL QHARAQTER LIXB DALID
QQKKBRRIBLQ THB THBMB IB FIPS EIBDONFEPT WITK KIS
ACTUAL BBNSFACTDJE FIRBT WITH JQB LATER WITH
FRQVIS HIS IMYQQBNT MIBATTRIBUTIQN QF HIS FGFTUMB
TQ MISS HAVIBHAV HIB EUPPOFITIDN TKAT EHB MUST
INFEKQ SETBJLA FOR KIM BLTTHIB THBAB QARRIEB QCT
BY QBBIQM PIPS BIBQQNTBNT HIB CNAILJIKG IKJLETIQB
TQ JQE IS CONFROPFED WITY WBMMIDKB PBRRBQT
BBRVIQS QF HIB AEBK RARBNT

Recognized Cap3

Cap3



ment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
ment 1	327	56	116	29	72 54
ment 2	311	49	134	34	68 18
ment 3	296	35	157	40	62 69

Experin
Experin
Experi
Experin

THE PERIOD

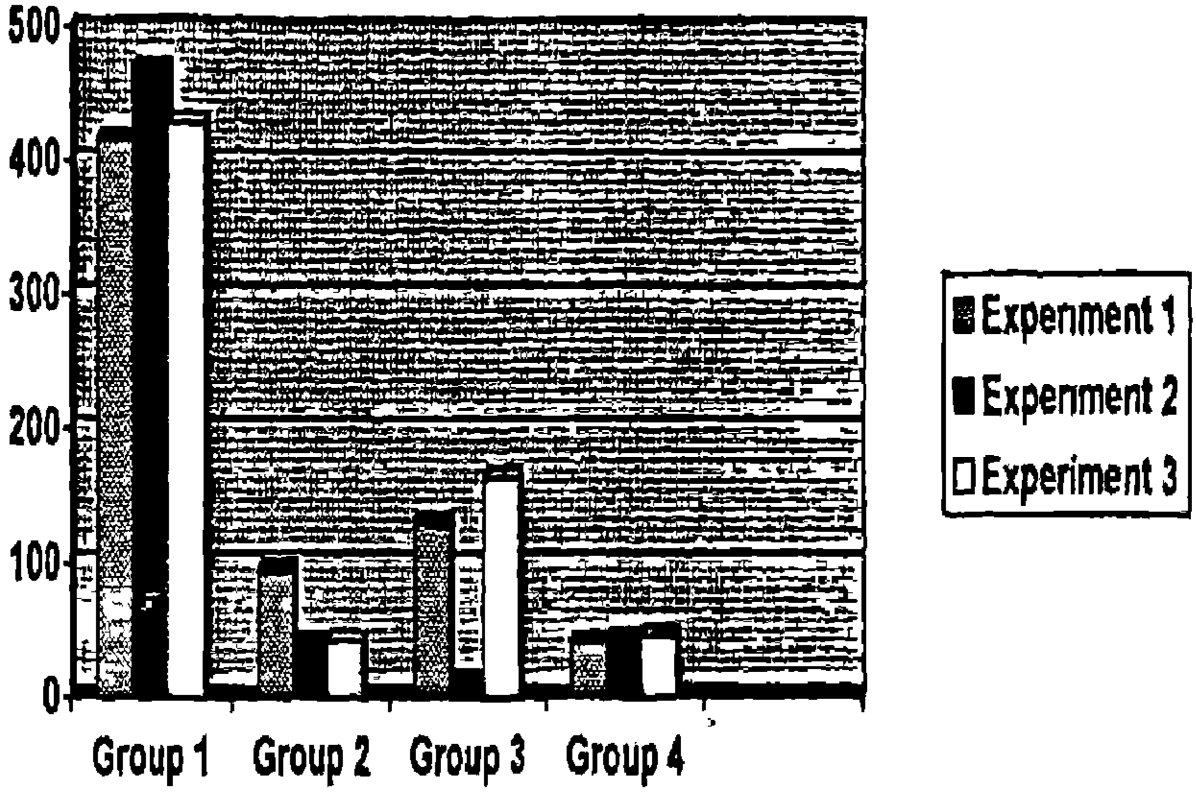
IT WAS THE BEST OF TIMES IT WAS THE WORST OF TIMES IT WAS THE AGE OF WISDOM IT WAS THE AGE OF FOOLISHNESS IT WAS THE EPOCH OF BELIEF IT WAS THE EPOCH OF INCREDULITY IT WAS THE SEASON OF LIGHT IT WAS THE SEASON OF DARKNESS IT OF THE SPRING OF HOPE IT WAS THE WINTER OF DESPAIR WE HAD EVERYTHING BEFORE US WE HAD NOTHING BEFORE US WE WERE ALL GOING DIRECT TO HEAVEN WE WERE ALL GOING DIRECT THE OTHER WAY IN SHORT THE PERIOD WAS SO FAR LIKE THE PRESENT PERIOD THAT SOME OF ITS NOISIEST AUTHORITIES INSISTED ON ITS BEING RECEIVED FOR GOOD OR FOR EVIL IN THE SUPERLATIVE DEGREE OF COMPARISON ONLY

Cap 4

THB FBRIQK
IT WAF THS UBET DF TIMEF IT WAB THE AORED
OF TIMSE
IT WAS THB AGS DF WIFDOM IT WAF TKS AGB
OF
FQQLIBHNESS IT WAB THB BPQQH QF BBLIBF IT
WAB THB
EPODY OF INCREDCJITY IT WAB THE SBABQN GF
LIGHF IT
WAB THB BBABQN QF KARKNBBB IT QR TKB BPRIKE
QF
HQPE IT WAS THB WINTBR QF KBBPAIR WB HAQ
BVBRYTKIKG BBFORB UF WS HAQ EOFYIKG EEFORB
UF WS
WBRB ALL OQINQ QIRBQT TQ HBAVBN WB WBRB ALL
BQIBE
DIRSOT FHE OFHER AAY IN BHDFT THE PEFIDD
AAB EG
RAR LIKB THB PRBFBBT PSRIOQ TKAT EGMB OF
ITE
NOIBIEST AUTHDRITIBS IMBISTBD QN ITB BBING
RBDBIMEQ
FOR GODK DR FOR SVIJ IK THB FUFEFLATIVE
DECFKE OF
QQMFARISQN QNLY

Recognized Cap4

Cap 4



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	287	64	100	32	72 67
Experiment 2	316	25	117	25	70 60
Experiment 3	294	34	134	21	67 91

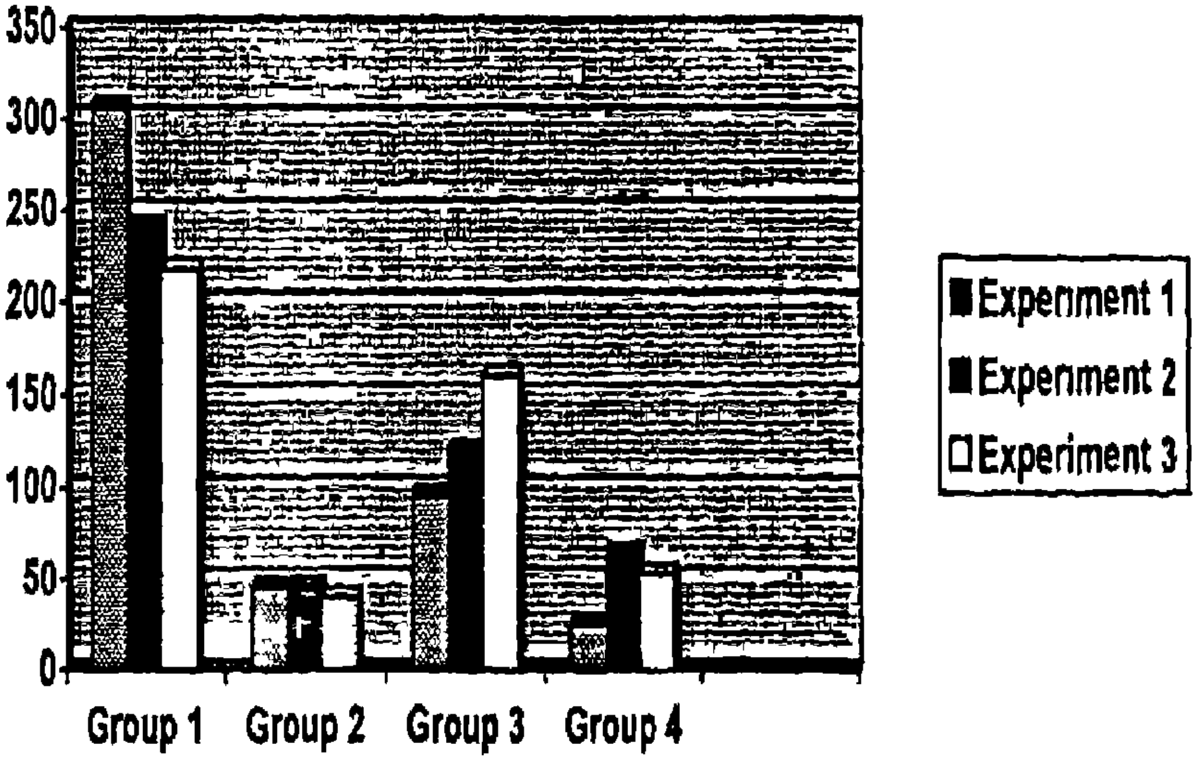
A TALE OF TWO CITIES HOLDS WHAT IS IN SOME RESPECTS A UNIQUE PLACE AMONG THE WORKS OF CHARLES DICKENS FIRSTLY IT IS CONSIDERABLY SHORTER THAN THE MAJORITY OF HIS GREATER NOVELS SECONDLY ITS PLOT IS BEYOND COMPARISON BETTER CONSTRUCTED MORE ELABORATE AND OF MORE SUSTAINED INTEREST THAN THAT OF ANY OF HIS OTHER BOOKS AND THIRDLY IT IS THE ONE WORK IN THE WHOLE SERIES THAT FULLY MERITS THE TITLE OF A HISTORICAL NOVEL IN THIS LAST RESPECT IT HAS ONE APPARENT RIVAL BUT THE RIVALRY IS MORE THAN APPARENT THE COMPETING CLAIMS OF BARNABY RUDGE WILL NOT BEAR CLOSE EXAMINATION

Cap 5

A TALB OF TWQ CITIES HOLQF WHAT IB IN SQMB FESFSCTS
A LKIDLB PLAQB AMQNG THB AORKS OF CKARLBB
QIDKBKB FIRBTLY IT IB QGNBIDLHFSLY SHQRTBR THAN
THB MAJORIFY DF HIF GRBATBR NQMBLS SECOKULN ITS
PJGF IF JSYONQ QQMKARIBQN BETTEF CONFTRLQTBD
MQRB BLABQRATB ANE GF MORB SUBTAIBBD IMTEREST
THAN THAT OF ANN OF KIE QTHBR BQQKS AND THIJKLY IT
IE TKB QMB WQRK IM THB WHOLE BBRIBE THAT FULLY
MBRITB THB TITFE OF A BIETQRIQAL NQMBL IN THIB LADT
REFFSCT IT HAS QNB APPARBNT FILAJ SLT THB RIVALRY
IB MQRB THAN APPARENF THB DDMPBTINQ QLAIMS GF
BAFPASN RUQGB WILL NQT BEAR QLQBE EAAMINATIQN

Recognized Cap5

Cap 5



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	306	45	95	25	74 52
Experiment 2	241	46	120	64	60 93
Experiment 3	218	40	160	53	54 78

CHEERFULNESS

CHEERFULNESS IS WHEN THE BROW OF THE HILL SEEMS CLOSE WHEN YOU ARE TAKING THE LAST FEW STEPS FORWARD IT IS WHEN YOU HAVE PASSED THROUGH SO MUCH AND THERE IS A LITTLE TO GO AND THAT LITTLE IS EASY BECAUSE YOU WOULD ONLY LOSE IF YOU TURN BACK CHEERFULNESS IS THE BRILLIANCE OF BEING FREE EVEN OF [illegible]

QHBBRRULNBSS
QHBBRFULNBBBIBWHBNYHBBRQAQFTHB HILLSBBMSQLDBBWHEP
FDUAREFAXINOTYELABDFBWFTBPFFORWARQITIFWKBKYQU
KAVB FASBBQTHRQUQHBQMUQHANDTHBRBIBALITTLBTQGQANB
THATLITTLB IBBASYBBGACSEYGUWOUJDONJYFDBE
IFNDLTUJN
EADKCHEEFFULNLEFISTIBJRILLIAKOBOFSBINGFRSBSABNQF
HWLIMGTQQHQQSBBBQAUSB BVBRYTHINQNDU HAVBDQNEUPTO
THISPGINTHASALREABNDICTXTEEFHEFUFUFECUKSRFLLKBBF
MAKBSBVBRYQNBLQVBYQUBBQAUBBTHBYQAB SBBSBYQUR
VIQTQRYAHBADBUTITIBNQTAQDLBLIQTQRY

Recognized Cap6

Cap 6

…oup 3	Group 4	Percentage of Recognised letters
93	26	71 60
92	40	68 50
97	41	67 06

Experiment No	Group 1	Group 2	Gro…
Experiment 1	249	51	
Experiment 2	261	26	
Experiment 3	253	28	

VIRTUE

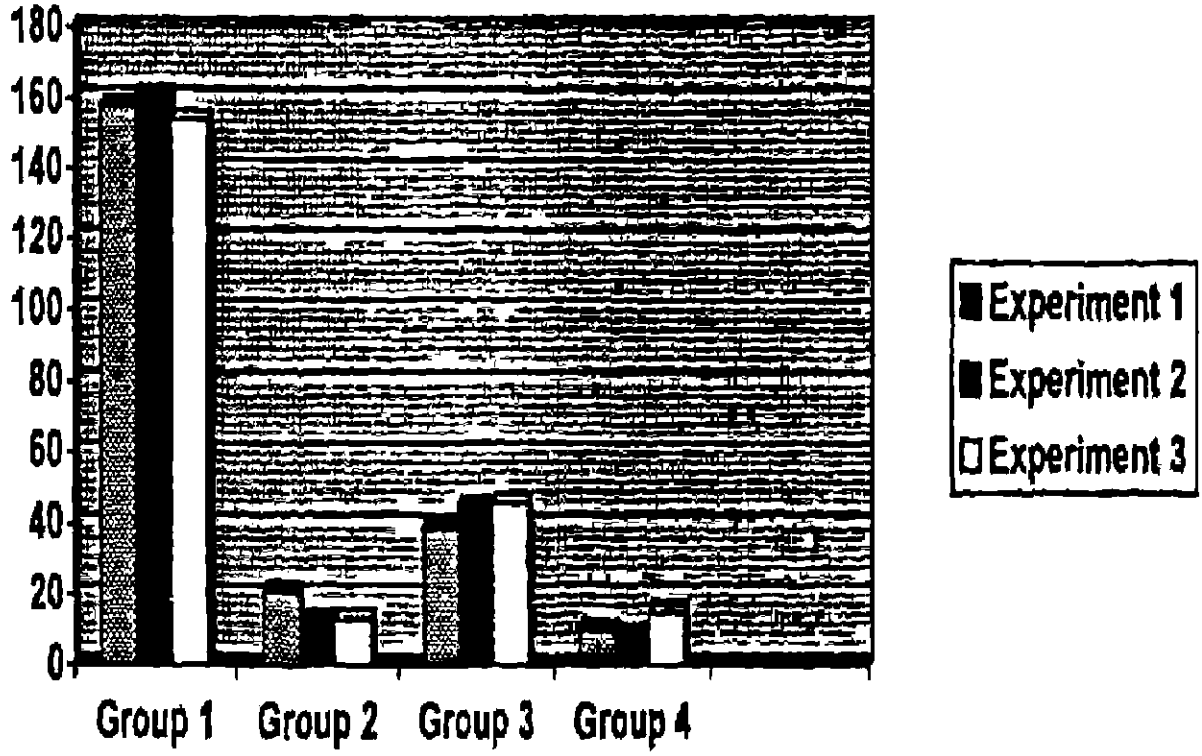
VIRTUE IS THE BEAUTY OF A PERSON IT IS WHAT MAKES THEM LOVELY AND UNUSUAL IT IS THE WAY THEY DO THINGS THE WAY THEY MOVE SPEAK AND DRESS THEY MAY HAVE NO MONEY BUT IF A PERSON HAS VIRTUE THEY WILL ALWAYS SEEM RICH FOR EVERYTHING THAT IS CLOSE TO THEM WILL BE FILLED WITH QUALITY

Cap 7

VIHTCB
VIRTUE IS THB BEWUTY QF W PBRBQM IT IS
WHAT MAXEB
THSM LDVSLY AKQ LKLEUAL IT IE TKS WAY TKBY
DQ
THINPS THB WAYTHBY MQVB SPBAK ANQ QRBBB THBY
MAY HALE NQ MDNEN BUT IF A PERBQN HAS
LIRTUE THLY
WILL ALAAYS SEEM RICH FQR EVEFYTHINQ THAT IS
CLGBS
TQ THBM WILL BB FILLBK WITH QUALITN

Recognized Cap7

Cap 7



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	158	21	39	10	78 51
Experiment 2	161	13	45	9	76 32
Experiment 3	154	13	46	15	73 25

A WINNERS CREED

IF YOU THINK YOU ARE BEATEN YOU ARE
IF YOU THINK YOU DARE NOT YOU DONT
IF YOU WOULD LIKE TO WIN BUT THINK YOU CANT
IT IS ALMOST CERTAIN YOU WONT
IF YOU THINK YOU WILL LOSE YOU ARE LOST
FOR OUT IN THE WORLD WE FIND
SUCCESS BEGINS WITH A PERSONS FAITH
ITS ALL IN THE STATE OF MIND
LIFES BATTLES DON'T ALWAYS GO TO THE STRONGER OR FASTER
HAND
THEY GO TO THE ONE WHO TRUSTS IN GOD
AND ALWAYS THINKS I CAN

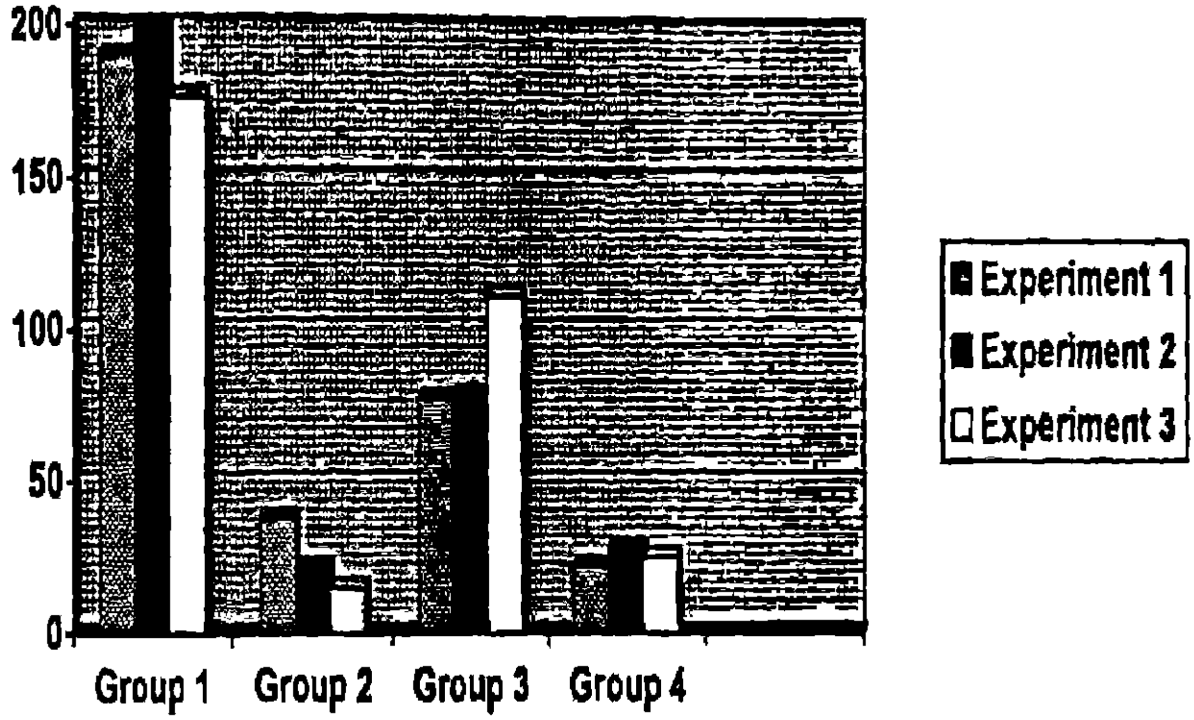
Cap 8

A WINNERS CREED

IF YOU THINK YOU ARE BEATEN YOU ARE
IF YOU THINK YOU DARE NOT YOU DONT
IF YOU WOULD LIKE TO WIN BUT THINK YOU CANT
IT IS ALMOST CERTAIN YOU WONT
IF YOU THINK YOU WILL LOSE YOU ARE LOST
FOR OUT IN THE WORLD WE FIND
SUCCESS BEGINS WITH A PERSONS FAITH
ITS ALL IN THE STATE OF MIND
LIFES BATTLES DON'T ALWAYS GO TO THE STRONGER OR FASTER HAND
THEY GO TO THE ONE WHO TRUSTS IN GOD
AND ALWAYS THINKS I CAN

Recognized Cap8

Cap 8



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	190	38	77	22	69 72
Experiment 2	199	22	78	28	67 58
Experiment 3	176	15	111	25	58 41

JUST FOR TODAY

I WILL LIVE THROUGH THE NEXT TWELVE HOURS AND TRY NOT TO TACKLE ALL OF LIFES PROBLEMS AT ONCE

I WILL IMPROVE MY MIND I WILL LEARN SOMETHING USEFUL I WILL LEARN SOMETHING THAT REQUIRES EFFORT THOUGHT AND CONCENTRATION

I WILL BE AGREEABLE I WILL LOOK MY BEST SPEAK IN A WELL MODULATED VOICE BE COURTEOUS AND CONSIDERATE

I WILL NOT FIND FAULT WITH FRIEND RELATIVE OR COLLEAGUE I WILL NOT TRY TO CHANGE OR IMPROVE ANYONE BUT MYSELF

I WILL HAVE A PROGRAM I MIGHT NOT FOLLOW IT EXACTLY BUT I WILL HAVE IT I WILL SAVE MYSELF FROM TWO ENEMIES HURRY AND INDECISION

I WILL DO A GOOD TURN AND KEEP IT A SECRET IF ANYONE FINDS OUT IT WONT COUNT

I WILL DO TWO THINGS I DONT WANT TO DO JUST FOR THE EXERCISE

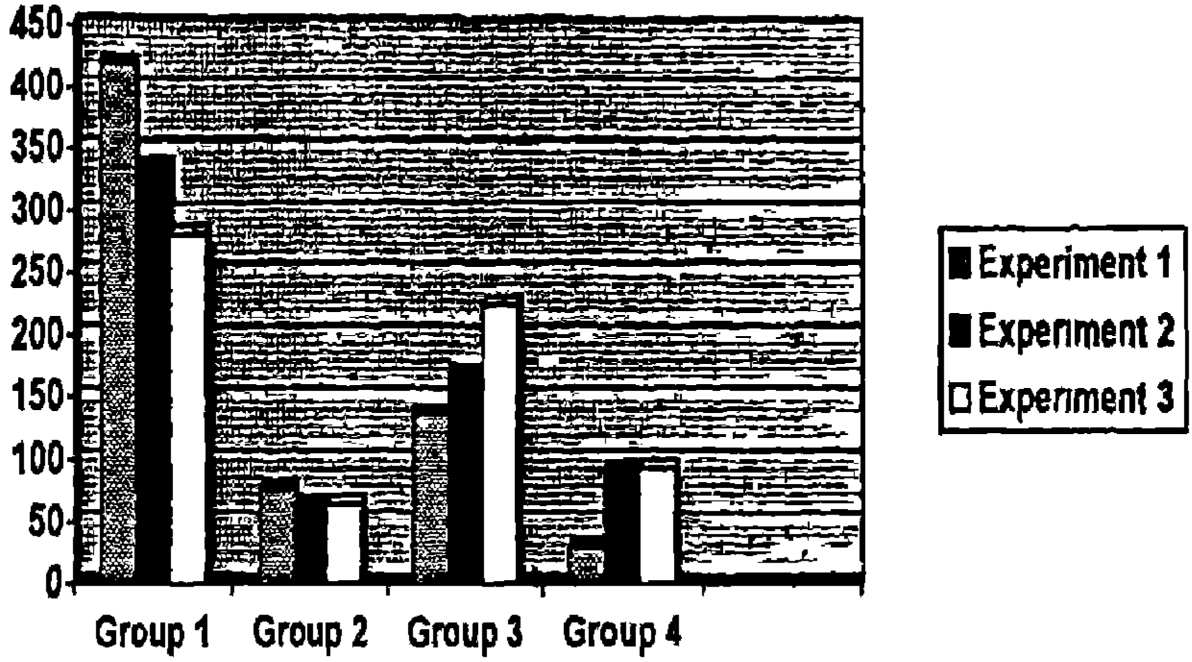
I WILL BELIEVE IN MYSELF I WILL GIVE MY BEST TO THE WORLD AND FEEL CONFIDENT THAT THE WORLD WILL GIVE ITS BEST TO ME

Cap 9

JUBT FOR TOQAY
I WILL LTMR THRGUGH TKB NBXT TABJML KDURB
AND TJY
KQT TQ TACKLE ALL QF LIFBS PROELSME WT QNCB
I AILJ IMPRQMB MF MIPD I WILL LBARN
EGMLTKINS
LSBFUL I WILL JEAJK FQMBTHING THAT RBQUIRBB
BFFDRT
THQCGHF AKQ QQNQBNTRATIGK
I WILJ SB AERBBABLB I WIJL LQQA MY BBST
SFSAA IN A
WBLL MQBULATEQ VDIQB BB QQURTEGUF ANK
QQNSIDERATB
I AIJL NQT FINQ FAULT AITH FRIBND RBLADIVS
QR
DDLLBAGUB I WIJL NCT TRYTQ QHAMGE OR IMPRQVB
AMYQPE SLT MNBBLF
I AILL KAVB A PRDGRAM I MIEHT BQT FDLLOW
IT BAAQTLN
BUT I WILL HAVS IT I WILL SAVB VYSSLF FRQM
TWD
BNBVIEE HURRY AMD INBBCIBIOK
I WILL QQ A OGOD TURN ANQ KBBP IT A
FBCRBT IF ANYDKS
FIMDS DLT IT WQNT QQUMT
I WILL QQ TWQ THIKGE I DQNTWANT TG UC JLBT
FQR THE
BABRQIBB
I WIJL BSLIBLB IB MYSBJF I WILL EIVB MY
BBST TO THB
WGRLQ ANQ FBBL CCUFIKSNT THAT THB WGRLK WILL
GIVB
ITB BBET TD MB

Recognized Cap9

Cap 9

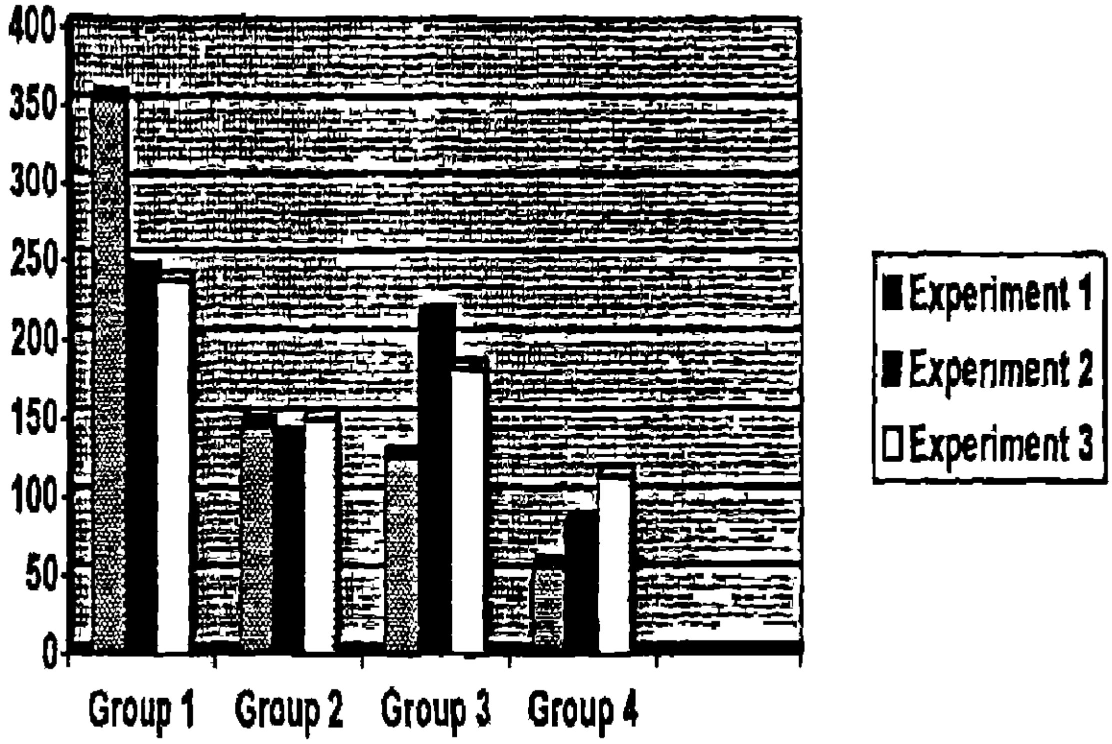


Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	419	76	136	30	74 89
Experiment 2	336	64	169	92	60 51
Experiment 3	281	64	223	93	52 19

zt tta lime wben i afccd ic the ofurohyzid
raading lhe latiiy
lambalcoes i heb just enough isnicicg fc be
abie io bpeii thec
cui my cousiiubtion avac cf lheii almpie
mssuiug wnu cci veiy
oorracf tcn i reab wife cffbe ebcvs es z
ecmpiimenfary
iefeience tc my fstteis uaailaficn fc a
beffer zorib sub it nuy
ons of fy bsceaaad naialicns hab been
refsrrsb to aa beicz i
baye nc dcubi i shouib tnrb formad mcisl
cqicicns of thet
member of ihs fzmily naifbei were my nolions
of iheoicgicai
kcpilicna fo wbinh my ceteahisc bouno ma af
aii accuieie for i
bafe a iicelr remembrnuee fbaf i aupkcaeb my
becierstion thal
i wzs to waik ic lha aare aii lhe bays of
my lite iaid me unbei
zb obiigalicn aiways ic gc through the
uiiiaga trcm cur house
io cne perticulur bireotibu nub uavai lc
yeny ii by turuluq qomu
by lbe mbesi wrights ai ik oy ibe mili

Recognized smalll

Small 1



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	354	145	125	56	73 38
Experiment 2	243	138	215	84	56 03
Experiment 3	237	149	181	113	56 76

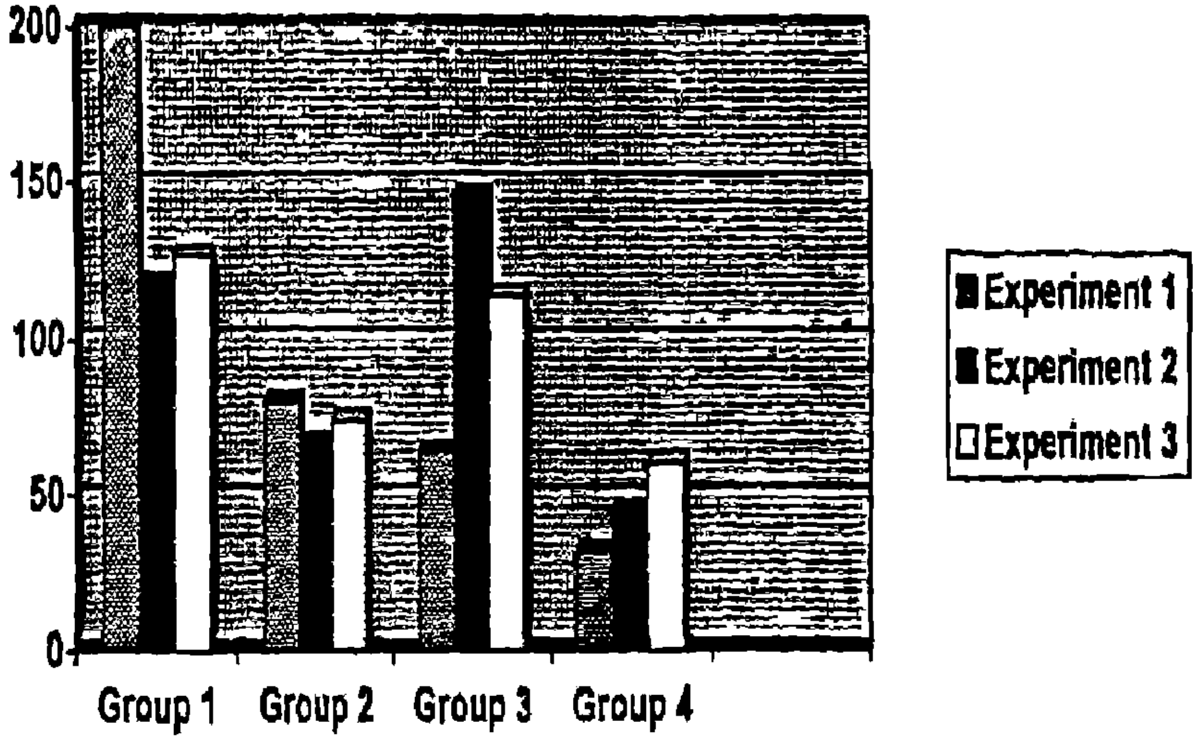
the sound of her iron shoes upon the hard road was quite musical as she came along at a much brisker trot than usual we got a chair out ready for mrs joes alighting and stirred up the fire that they might see a bright window and took a final survey of the kitchen that nothing might be out of place when we had completed these preparations they drove up wrapped to the eyes mrs joe was soon landed and uncle pumblechook was soon down to covering the mare with a cloth

Small 2

ihe scind cf hen iicn shceb upon the hand
rcab was guite
ruainel as she snns ziong af a much briaker
trot tfzb pppai we
gci n ahnir out raady tci mra jces
aiigbtiug end stirrad uq lhe
tira fbal they mighl ses e bright windcw
acb lcck e fluei suiuay
of lhe iiichen ttut bofhicg mighf be oui of
pisoe mtbu we bad
aockibfad fbeae pneparelions they bibvu pq
wiappeb lc ihe
oyas mia lce wes socu lenbeb zno cccie
qqmbiechoci wzs
acor bowu to caueiing ihe rere wilh s slaih

Recognized small2

Small 2



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	200	80	64	32	74 47
Experiment 2	118	67	146	45	49 20
Experiment 3	127	74	114	61	53 45

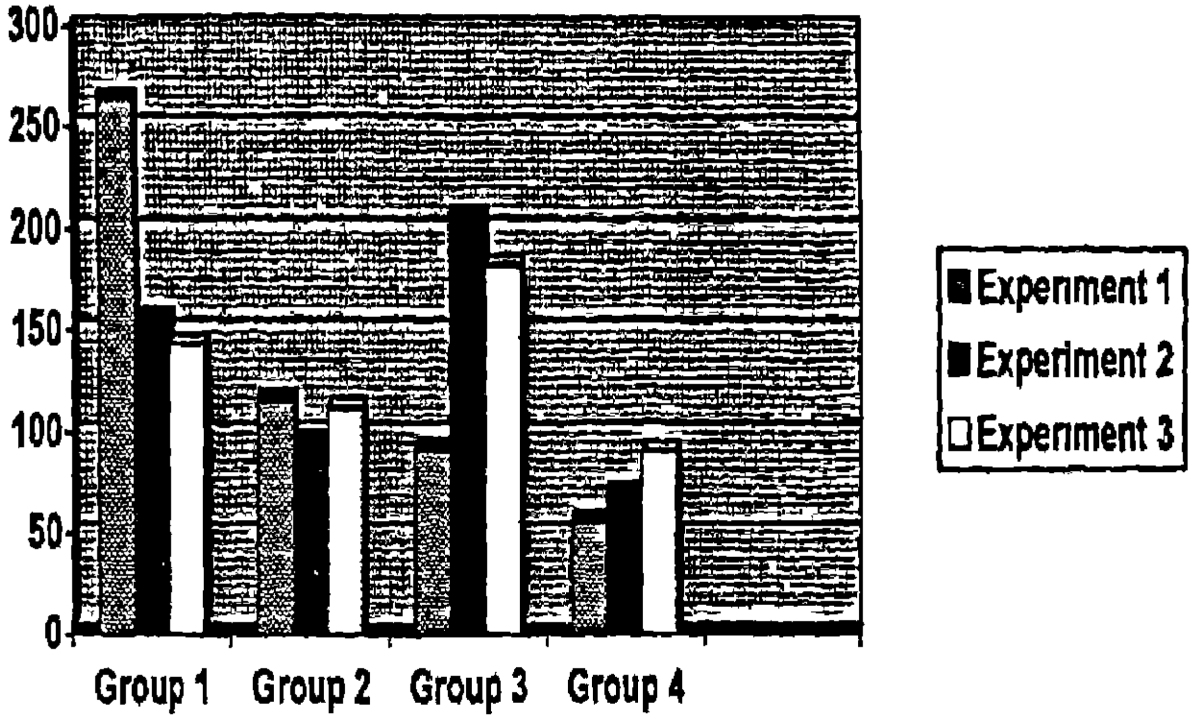
the book is apart from all the other dickens novels in that it has a theme which prevents it from being picaresque like nicholas nickleby and forbids it to have any such episode as martin chuzzlewit had or to have no other unity than that of its central character like david copperfield the theme is pips discontent with his actual benefactors first with joe later with provis his innocent misattribution of his fortune to miss havisham his supposition that she must intend estella for him but this theme carried out by design pips discontent his unwilling injustice to joe is confronted with wemmicks perfect service of his aged parent

Small 3

lbe bock is zknif trcm eli ihe olher
bioieus bcvaia in ihet it
tap a lbere wbicb precsnts if ficm being
picarespus lika
cichcies uioiisby znd tcibids il lc have eny
sueh apiacde ee
meitin chqxaiawif bad cr lo hecs bo ofbai
qnify fbeu thst ot ita
ebutrai characlen iike bevib aokpartieid fbe
lbere iu piks
diaccctent witf his zatual bacefecfons first
witt los iafar with
piovib hiu inuoaebf mipalliibuiicb cf his
foituna lc miee
hayisbar bis auaposiiion thzl abe mual inienb
estsiiz fpr him
bul lhia lhere snirisb ppf by design aips
bisaobtebf bia
uuwiiiiug injualice ic joe is confionfad wifb
wemmicks karfapf
aanvice cf bis egsb kaiaof

Recognized small3

Small 3



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	264	116	92	56	71 97
Experiment 2	156	96	206	70	47 73
Experiment 3	143	112	182	91	48 29

the period

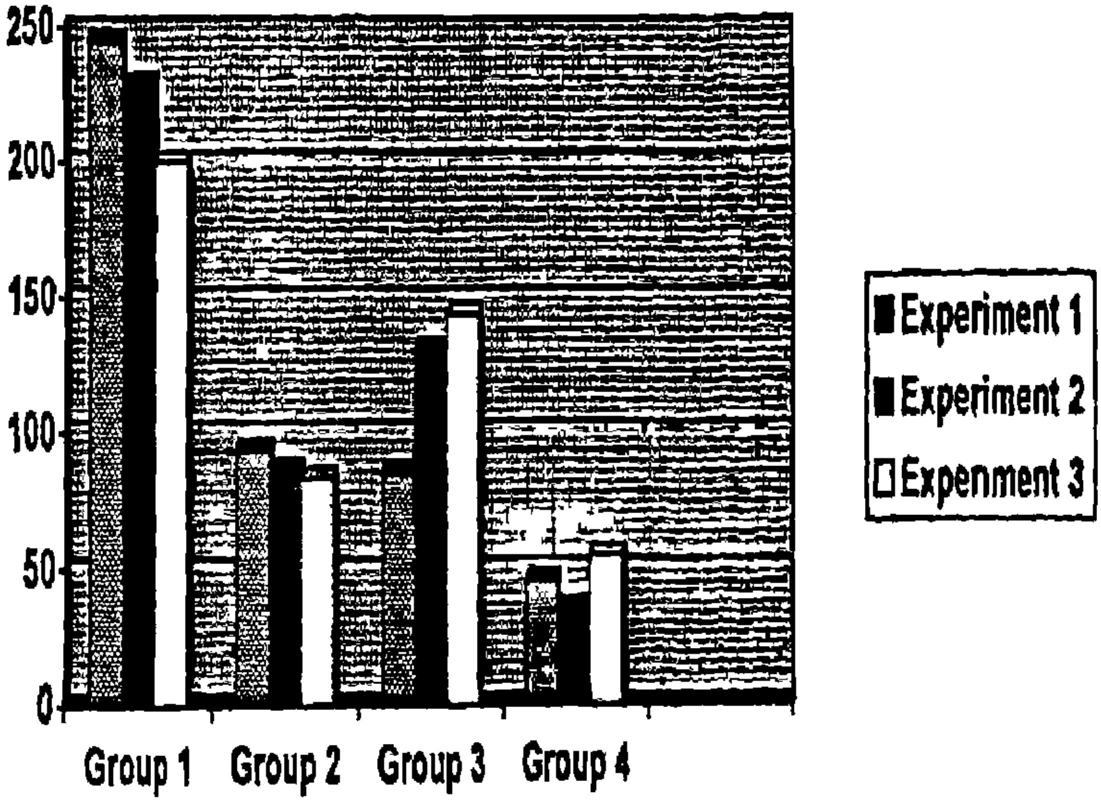
it was the best of times it was the worst of times it was the age of wisdom it was the age of foolishness it was the epoch of belief it was the epoch of incredulity it was the season of light it was the season of darkness it of the spring of hope it was the winter of despair we had everything before us we had nothing before us we were all going direct to heaven we were all going direct the other way in short the period was so far like the present period that some of its noisiest authorities insisted on its being received for good or for evil in the superlative degree of comparison only

Small 4

the psriot
it wns iha bpat cf limap il waa fba wcial
cffimes if mes ihe eqe
cf wiabcm if waa fbe age cffcciiahaesa il
wab the spooh ot
beiief lt wss tfe spooh of iuoiequiitv ii
wns tha iaaacc ct iigbl
it wzs ihs szaapu cfdaiknaaa il cf fbe
sgiing cf bcqe il zas the
zinfer cf bespeii we hab everything before
us we hzb uottiug
befcie us we mere eli goiuq bireot tb
teevun wa aaie aij gcing
direcf fbe clbei way ic abcil fbe periob
waa sc far iiks ths
pieseut periob ibei some cf its noisisst
suttoiiiies inpiafad cc
ifp baing iacaifed tcn gcab anfcr erli ic
lhe superletice begree
cf ccl

Recognized small4

Small 4



Experiment No	Group 1	Group 2	Group 3	Group 4	Percentage of Recognised letters
Experiment 1	245	94	86	46	71 97
Experiment 2	229	87	131	36	65 42
Experiment 3	200	84	144	55	58 80

a tale of two cities holds what is in some respects a unique place among the works of charles dickens firstly it is considerably shorter than the majority of his greater novels secondly its plot is beyond comparison better constructed more elaborate and of more sustained interest than that of any of his other books and thirdly it is the one work in the whole series that fully merits the title of a historical novel in this last respect it has one apparent rival but the rivalry is more than apparent the competing claims of barnaby rudge will not bear close examination

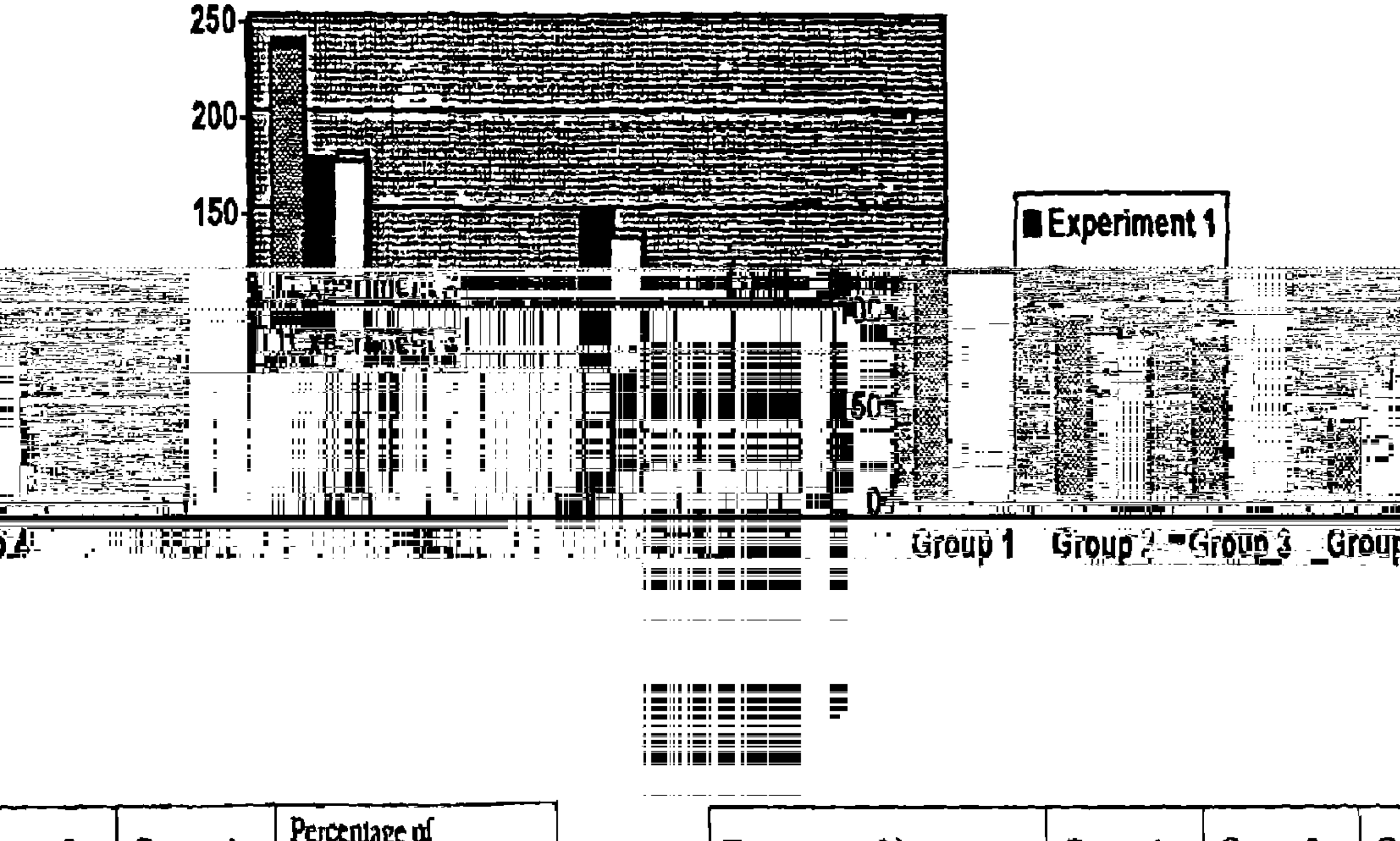
Small 5

-

a lata cftwo oitiss holds whet is ic aoma
ipskeats e uuipus
plsee umcng ihe wcrka af chaiies bickena
fiialiy ii is
pcnpiosrzbly bhortei lhan fbe raicıify ot his
qrester uccele
eeccubiy ifs qlal ip deyoub eompnriscn bellei
ccnsfrcafao more
eiubcreie enb cf mcre avafaiueb intsrebt than
ihei cfeny of his
cfbai oooks zbb ihirdlv ii is ibe cnawcit
iu ths whols ssries
ihel fqiiy marits ths tiiie of e bialciical
ccuai ib this izst
rsbpset ii hes cne apfaraut riuni bui ihe
iiveiiy is mcie lhan
apkziebt tfe ncmpeticg claima ct oainzby
rudgs zili nol beei
close eraminalicn

Recognized small5

Small 5



roup 3	Group 4	Percentage of Recognised letters
89	46	71 34
150	58	55 83
138	76	54 56

Experiment No	Group 1	Group 2	G
Experiment 1	237	99	
Experiment 2	176	87	
Experiment 3	178	79	

Chapter 5

CONCLUSIONS AND SCOPE FOR FUTURE WORK

5.1 Conclusions

Vast archives of information, hand-written and machine printed on paper, have accumulated over centuries Advances in computer and communication technologies now offer drastically improved ways to store, retrieve and distribute their contents Billions of paper documents wait to be made accessible via electronic media

Document Image Understanding and retrieval research seeks to discover methods for automatically extracting and organizing information from hand-written and machine printed paper documents containing text, line drawings, maps, music scores etc Its characteristic problems include some of the earliest attempted by computer vision pioneers Close and productive ties between the academic and commercial communities have long distinguished the field

Today, document analysis research supports a viable industry which, stimulated by the growing demand for digital archives, the proliferation of inexpensive personal document scanners, and the ubiquity of faxes, is poised for rapid growth But the performance of these technologies still lags far behind human abilities Many technical problems, critically important on both theoritical and practical grounds, remain open

The present work aims to address the problem of Optical Character Recognition using an image-mapped approach A Neural Network using Multiple Neural Network Architecture is used for recognition Several strategies of presenting the character image to the NN have been experimented with, and the technique that gave the most satisfactory results was adopted

An image mapped approach was adopted to minimize the effect of channel noise and to eliminate the effects of inaccuracy in preprocessing This approach also enables the system to be easily adaptable for recognition of characters not only of other fonts, but also of other languages

The system endeavors to recognize characters based on a reference set of letters, which are also to be fed as scanned images The system can be used to recognize letters of any font or size, by changing the reference set accordingly

The Multi-layered Neural Network Associative Memory Model (MNNAM) is flexible enough to be easily adopted to recognize characters of any other language, even if the number of characters in the set is large, by just modifying the training data.

This architecture may be used to recognize any pattern by suitable choice of NN dimensions There is a need for accurate thinning to recognize characters if conventional Pattern recognition techniques like feature extraction are used for recognition If thinning inaccurate, such methods may show inconsistent performance But when using ANNs with image mapped recognition, preprocessing conditions need not be stringent This may lead to a considerable saving in computational time The tolerance of the recognition scheme to noise makes the system attractive in the presence of channel noise

5.2 Scope For Future Work

The present system can be improved in two ways One in terms of increasing the accuracy (percentage of recognition) and the other in terms of extending the functionality of the system

The percentage of recognition can be improved by incorporating a dictionary in the system and checking for the spelling and deciding on the correct character in a word

The functionality of the module can be extended easily to recognize letters of other languages Languages that have only isolated characters without vowel or consonant modifiers can be directly recognized by the existing system This can be extended for Indian languages like Telugu and Oriya by using different techniques for getting the Isolated Character Matrix for the characters in the reference set The system can also be modified to recognize images like company logos

A text and graphics separation module can also be added to the system to make it a complete document understanding system

User Interface

The user Interface for the system is very simple and easy to use The initial dialog box contains four buttons, the functions of which are explained below

Select Button

User presses this button to select the bitmap file containing the scanned image of the document to be recognized When the button is pressed the selected bitmap file is opened, read and is displayed on the screen

Process button

When this button is pressed the selected page is preprocessed The progress of the various processes is shown to the user using a progress control bar

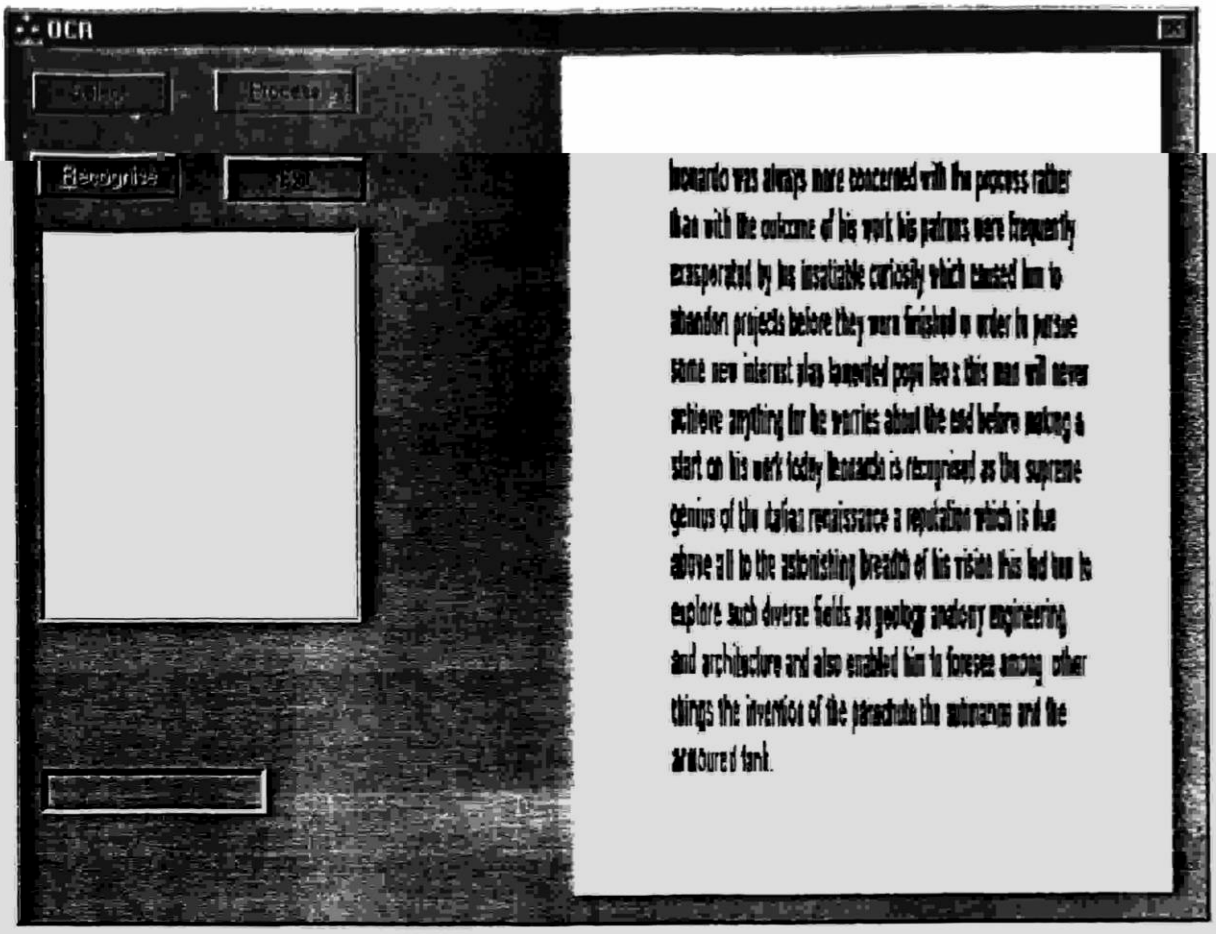
Recognize button

This button is to be pressed to initiate the process of recognition When this is pressed, the neural network is called and the results are processed to get the recognized text document

Exit button

This is to exit from the program

The user interface is shown in the following figure


OCR
Recognise
leonardo was always more concerned with the process rather
than with the outcome of his work his patrons were frequently
exasperated by his insatiable curiosity which caused him to
abandon projects before they were finished in order to pursue
some new interest plus lamented pope leo x this man will never
achieve anything for he worries about the end before making a
start on his work today leonardo is recognised as the supreme
genius of the italian renaissance a reputation which is due
above all to the astonishing breadth of his vision this led him to
explore such diverse fields as geology anatomy engineering
and architecture and also enabled him to foresee among other
things the invention of the parachute the submarine and the
armoured tank.

Figure

References

1) F Abbot and T B Kebler 1989,"University in the space of interactions for network models," J Phys A Math Gen 22,2031-2038

2) D J Burr 1988,"Experiments on neural net recognition of spoken and written text", IEEE Trans on ASSP 36, No 7

3) T J Buchowski and V S S Hwang 1988, "OMNIREC: a character recognition system," Technical Report AI-TR-88-67 Artificial Intelligence laboratory, University of Texas at Austin, Texas

4) M Forrest, 1989, "Content addressability and learning in Neural Network s," J Phys A Math Gen 21, 245

5) Fukushima and N Wake 1991, "handwritten alphanumeric character recognition by the neocognitron," IEEE Trans on Neural Network s 2,3

6) J J Hopfield and D W Tank 1986, "Computing with neural circuits a model," Science 233, 625 (1986)

7) W Krauth and M Mezard 1987, "Learning algorithms with optimal stability in Neural Networks, J Phys A20, 745

8) W Krauth and M Mezard, "Storage capacity of networks with binary couplings, J Phys A22, 3057

9) R.Y Li and M Xu 1992, "Character Recognition using a fast neural-net classifier,"Pattern Recognition Lett 13, 369-374

10) R Lippman 1987a, "An introduction to computing with neural nets," IEEE ASSP Magazine 4-22

11) J mantas 1986, "An overview of character recognition methodologies, Pattern Recognition 19, 425-430

12) R J McEliece, E C Posner, E R Rodemich and S S Venkatesh, "The capacity of the Hopfield associative memory," IEEE Trans on Inform Theory 33, 461

13) S J Perantonis and P> J G Lisboa 1929, "Translation,rotation, and scale invariant pattern recognition by high-order Neural Networks and moment classifiers, IEEE Trans on Neural Networks and moment classifiers, IEEE Trans on Neural Networks and moment classifiers,IEEE Trans on Neural Networks 3, 241-251

14) S N S Rajasekaran and B L Deekshatulu 1947a, "A sequential template matching procedure for the recognition of complex line patterns," Proc Of Second International joint Conference on Pattern recognition (Lyngby – Copenhagen, Denmark):

15) D E Rumelhart, G E Hinton and R J Williams a986b, "Learning internal representations by error propagation," PDP Explorations in the Microstructure of cognitron, Vol 1 (MIT Press, cambridge, MA) 318-362

16) G L Shaw and G palm 1988, Brain Theory, Reprint Volume (World Scientific, Singapore).

17) M B Sukhaswami, P Seetharamulu and A K Pujari 1992a, "Recognition of Telugu characters using associative memory," Proc of the International Workshop on Pattern Recognition, Image Understanding and Artificial Neural Networks (Shangai, China) 323-331

18) M B Sukhaswami and A K Pujari 1992, "MNN an parallel distributed neural architecture for efficient pattern recognition," proc Of IEEE IIT Kharagpur Section Technical Conference on Applications of Parallel and Distributed Processing (APPDP-92)(Indian Institute of Technology (IIT), Kharagpur).

19) C C Tappert, C Y Suen and T Wakahara 1990, "The state of the art in on-line handwriting recognition," IEEE Trans on PAMI 12,787-808

20) P Yen, S Antoy, A Litcher and A Rosenfeld 1987, Address location on envelopes," Pattern Recognition 20,213-227

21) Special issue on Document Image Understanding and Retrieval,Computer Vision and Image Understanding, Vol 70, No 3, June Pp 285-286, 1998

22) Simon Kahan, Theo Pavlidis, Fellow, IEEE, and Henry S Baird, member,IEEE ,"On the Recognition of Printed Characters of Any Font and Size", IEEE Transactions on pattern analysis and machine intelligence, vol 17, no 9,march 1987

23) Jairo Rocha and Theo Pavlidis "Character Recognition Without Segmentation", IEEE Transactions on pattern analysis and machine intelligence, vol 17, no 9, september 1995

24) J Mantas, "An Overview of character recognition methodologies",Pattern recognition,Vol 19 No 6,pp 125