Chapter 1

INTRODUCTION

People have always tried to develop machines which could do the work of a human being. The reason is obvious since for most of history, man has been very successful in using the machines developed to reduce the amount of physical labor needed to do many tasks. With the advent of the computer, it became a possibility that machines could also reduce the amount of mental labor needed for many tasks. Over the past fifty or so years, with the development of computers ranging from ones capable of becoming the world chess champion to ones capable of understanding speech, it has come to seem as though there is no human mental faculty which is beyond the ability of machines. Today, many researchers have developed algorithms to recognize printed as well as handwritten characters. But the problem of interchanging data between human beings and computing machines is a challenging one. In reality, it is very difficult to achieve 100% accuracy. Even humans too will make mistakes when come to pattern recognition.

The accurate recognition of typewritten text is now considered largely a solved problem in applications where clear imaging is available such as scanning of printed documents. Typical accuracy rates on these exceed 99%; total accuracy can only be achieved by human review. Other areas including recognition of hand printing, cursive handwriting, and printed text in other scripts especially those with a very large number of characters are still the subject of active research.

This project titled ‘Character Recognition System’ is an offline recognition system developed to identify either printed characters or discrete run-on handwritten characters. It is a part of pattern recognition that usually deals with the realization of the written scripts or printed material into digital form. The main advantage of storing these written texts in digital form is that, it requires less space for storage and can be maintained for further references without referring to the actual script again and again.

1.1 Image and Image Processing

Image is a two-dimensional function *f(x,y)*, where x and y are spatial coordinates and the amplitude f at any pair of coordinates *(x,y)* is called the intensity or gray level. When x, y, and f are discrete quantities the image is digital. ‘f’ can be a vector and can represent a color image, e.g. using the RGB model, or in general a multispectral image. The digital image can be represented in coordinate convention with M rows and N columns as in Figure 1.1. In general, the gray-level of pixels in an image is represented by a matrix with 8-bit integer values.

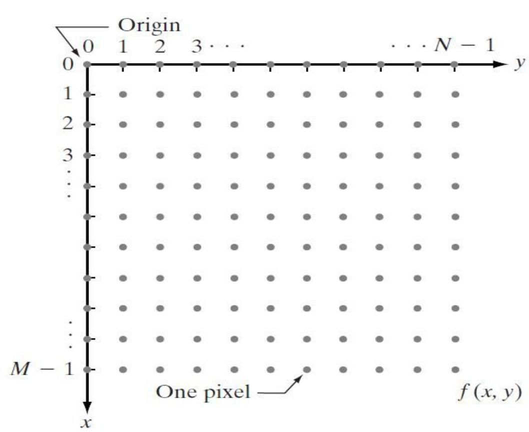


Figure 1.1: Coordinate convention used to represent an image

Image Processing is all about improvement of pictorial information for human interpretation and processing of image data for storage, transmission and representation for autonomous machine perception. Processing of image data enables long distance communication, storage of processed data and also for application which require extraction of minute details from a picture.

Digital image processing concerns with the transformation of an image to a digital format and its processing is done by a computer or by a dedicated hardware. Both input and output are of digital in nature. Some processing techniques tend to provide the output other than an image which may be the attributes extracted from the image, such processing is called digital image analysis. Digital image analysis concerns the description and recognition of the image contents where the input is a digital image; the output is a symbolic description or image attributes. Digital Image Analysis includes processes like morphological processing, segmentation, representation & description and object recognition (sometimes called as pattern recognition).

Pattern recognition is the act of taking in raw data and performing an action based on the category of the pattern. Pattern recognition aims to classify data (patterns) based on the information extracted from the patterns. The classification is usually based on the availability of a set of patterns that have already been classified or described. One such pattern is Character. The main idea behind character recognition is to extract all the details and features of a character, and to compare it with a standard template. Thus it is really necessary to segment these characters before proceeding with the recognition techniques. To achieve this, the printed material is stripped into lines, and then into individual words. These words are further segmented into characters.

1.2 Characters - An overview

Characters in existence are either printed or handwritten. The major features of *printed characters* are that they have fixed font size and are spaced uniformly and they do not connect with its other neighboring characters. Whereas handwritten characters may vary in size and also the spacing between the characters could be non-uniform. Handwritten characters can be classified into, discrete characters and continuous characters. The different types of handwritten characters are shown in Figure 1.3.

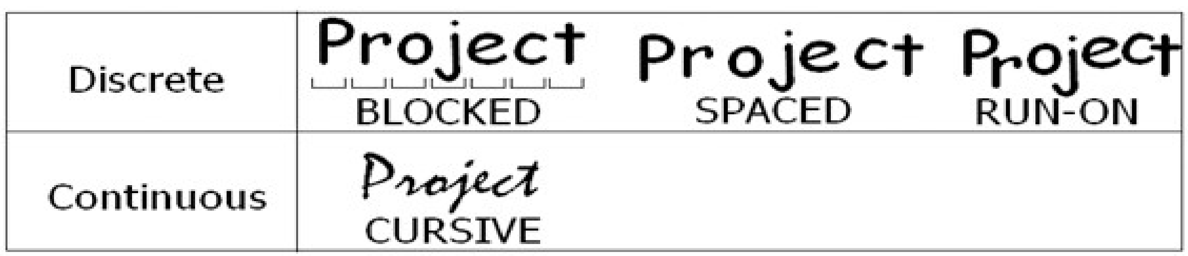


Figure 1.3: Handwritten character styles.

Processing of printed characters is much easier than that of handwritten characters. By knowing the spaces between each character in printed format, it is easy to segment the characters. For handwritten characters, connected component analysis has to be applied, so that all the characters can be extracted efficiently.

Although there are 26 characters in English language, it is observed that both uppercase and lowercase letters are utilized during the construction of a sentence. Thus, it is necessary to design a system which is capable of recognizing a total of 62 elements (26 lowercase characters + 26 uppercase letters + 10 numerical).

1.3 Literature Review

In 1929 Gustav Tauschek obtained a patent on OCR in Germany, followed by Paul W. Handel who obtained a US patent on OCR in USA in 1933. In 1935 Tauschek was also granted a US patent on his method. Tauschek's machine was a mechanical device that used templates and a photo detector.

In 1949 RCA engineers worked on the first primitive computer-type OCR to help blind people for the US Veterans Administration, but instead of converting the printed characters to machine language, their device converted it to machine language and then spoke the letters. It proved far too expensive and was not pursued after testing

In 1950, David H. Shepard, a cryptanalyst at the Armed Forces Security Agency in the United States, addressed the problem of converting printed messages into machine language for computer processing and built a machine to do this, reported in the Washington Daily News on 27 April 1951 and in the New York Times on 26 December 1953 after his was issued. Shepard then founded Intelligent Machines Research Corporation (IMR), which went on to deliver the world's first several OCR systems used in commercial operation.

In 1955, the first commercial system was installed at the Reader's Digest. The second system was sold to the Standard Oil Company for reading credit card imprints for billing purposes. Other systems sold by IMR during the late 1950s included a bill stub reader to the Ohio Bell Telephone Company and a page scanner to the United States Air Force for reading and transmitting by teletype typewritten messages. IBM and others were later licensed on Shepard's OCR patents.

1.4 Scope of the Project

On-line systems for recognizing hand-printed text on the fly have become well-known as commercial products in recent years. Among these are the input devices for personal digital assistants such as those running Palm OS. The Apple Newton pioneered this product. The algorithms used in these devices take advantage of the fact that the order, speed, and direction of individual lines segments at input are known. Also, the user can be retrained to use only specific letter shapes. These methods cannot be used in software that scans paper documents, so accurate recognition of hand-printed documents is still largely an open problem. Accuracy rates of 80% to 90% on neat, clean hand-printed characters can be achieved, but that accuracy rate still translates to dozens of errors per page, making the technology useful only in very limited applications.

Chapter 2

CHARACTER RECOGNITION

"Character Recognition" is an offline recognition system developed to identify either printed characters or discrete run-on handwritten characters. It is a part of pattern recognition that usually deals with the realization of the written scripts or printed material into digital form. The main advantage of storing these written texts in digital form is that, it requires less space for storage and can be maintained for further references without referring to the actual script again and again. Character recognition has wide applications such as in postal services to sort the mails according to their destination using the addresses that are written on the envelope, in restoring old manuscripts, in digital signature verification and much more.

2.1 Classifications of Character Recognition systems

Character recognition is a process, which associates a symbolic meaning with objects (letters, symbols and numbers) that are on an image, *i.e.,* character recognition techniques associate a symbolic identity with the image of a character. Mainly, [2] character recognition machine takes the raw data that further implements the *preprocessing* of any recognition system.

Character recognition is an extremely large field which can be divided generally into two fields:

- On-line character recognition and
- Off-line character recognition.

On the basis of that data acquisition process, character recognition system can be classified into following categories as shown in figure 2.1

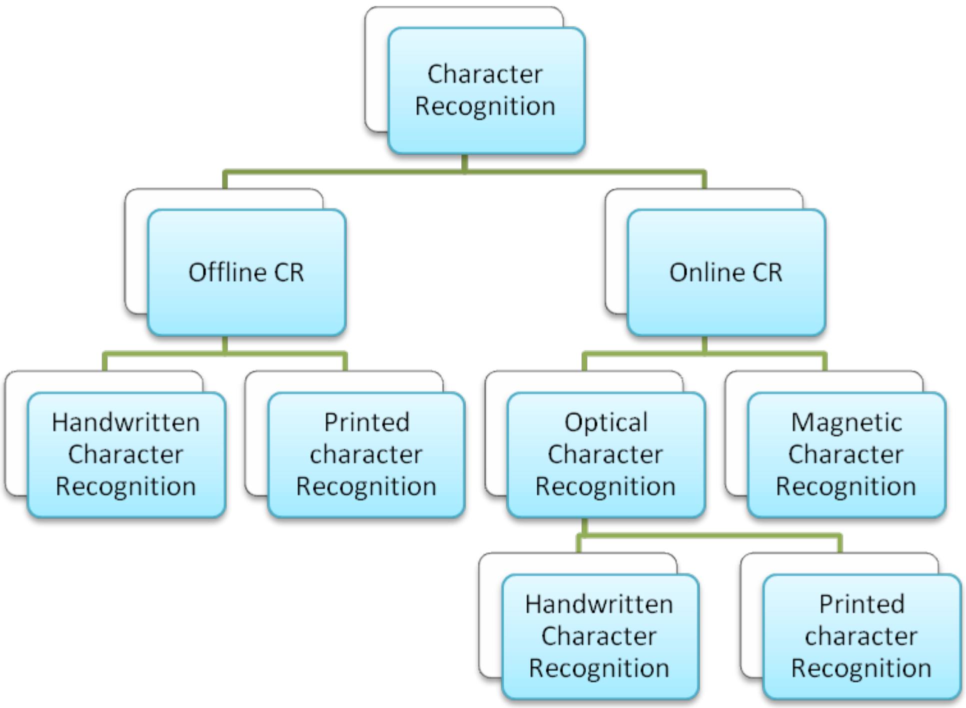


Figure 2.1: Classification of Character recognition systems

2.1.1 On-line character recognition

On-line recognition is the process of recognizing the characters as they are being written. Also online character recognition has real time contextual information [3]. Examples of systems that employ on-line recognition include the Apple Newton, Palm Pilot, Touch screen mobiles. In case of online handwritten character recognition, the handwriting is captured and stored in digital form via different means. Usually, a special pen (Stylus) is used in conjunction with an electronic surface. As the pen moves across the surface, the two- dimensional coordinates of successive points are represented as a function of time and are stored in order.

2.1.2 Off-line character recognition

On the other hand, off-line recognition is a system that recognizes by capturing an image of the characters or handwritten text that are to be recognized. Off-line recognition systems' potential lies in fields such as document processing, mail direction, and cheque verification. Off-line handwriting recognition refers to the process of recognizing words that have been scanned from a surface (such as a sheet of paper) and are stored digitally in gray scale format. After being stored, it is conventional to perform further processing to allow superior recognition but offline data does not support for real time contextual information. This difference generates a significant divergence in processing architectures and methods.

The offline character recognition can be further classified into two types:

i. Magnetic Character Recognition (MCR)
ii. Optical Character Recognition (OCR)

i. Magnetic Character Recognition (MCR)

In MCR, the characters are printed with magnetic ink. The reading device can recognize the characters according to the unique magnetic field of each character. MCR is mostly used in banks for cheque authentication service and also for updating entries in the transaction statements.

ii. Optical Character Recognition (OCR)

OCR deals with the recognition of characters acquired by optical means, typically a scanner or a camera. The characters are in the form of pixilated images, and can be either printed or handwritten, of any size, shape, or orientation. OCR can be subdivided into handwritten character recognition and printed character recognition. Handwritten Character Recognition is more difficult to implement than printed character recognition due to diverse human handwriting styles and customs. In printed character recognition, the images to be processed are in the forms of standard fonts like Times New Roman, Arial, Courier, etc.

2.2 Steps in Character Recognition

The general steps that are involved in character recognition systems are,

1. Image acquisition
2. Preprocessing
3. Segmentation
4. Character recognition

2.2.1 Image Acquisition

This is the stage where the image under consideration is taken. In the case of online recognition system a specialized hardware is implemented as explained earlier whereas for offline systems, the images are obtained either through a scanner or a camera. Whenever an image is acquired, there will be some variations in the intensity levels along the image. Also noise gets added to the image. Hence preprocessing is required for adjusting the intensity levels and to denoise the image.

2.2.2 Preprocessing

Preprocessing is the most important part of a better performing recognition system. In this stage, the acquired image is processed to remove any noise that may have incurred into the image during the time of acquisition or during the time of transmission. A colored image then it will be converted to a gray image before proceeding with the noise removal procedure. The denoised image is then converted to a binary image with suitable threshold.

2.2.3 Segmentation

Segmentation refers to a process of partitioning an image into groups of pixels which are homogeneous with respect to some criterion. Segmentation algorithms are area oriented instead of pixel oriented. The result of segmentation is the splitting up of the image into connected areas. Thus segmentation is concerned with dividing an image into meaningful regions. Image segmentation can be broadly classified into two types[3]

i. Local Segmentation: It deals with the segmenting sub images which are small windows on a whole image.

ii. Global segmentation: It deals with the images consisting of relatively large number of pixels and makes estimated parameter values for global segments more robust.

For character segmentation, first the image has to be segmented row-wise (line segmentation), then each rows have to be segmented column-wise (word segmentation). Finally characters can be extracted using suitable algorithms such as edge detection technique; histogram based methods or connected component analysis method.

Connected component analysis is an algorithmic application of graph theory, where subsets of connected components are uniquely labeled based on a given heuristic. Connected component analysis is used in computer vision to detect connected regions in binary digital images, although color images and data with higher-dimensionality can also be processed. When integrated into an image recognition system or human-computer interface, connected component labeling can operate on a variety of information.

2.2.4 Character Recognition

Recognition is the last step in the character recognition process. Character recognition is done using specific algorithms which requires database or templates to be stored and then used to recognize the segmented characters. Database is nothing but the collection of the templates of all the characters of different styles and fonts.

2.3 Database creation

Database is like the heart for the recognition system. It is the collection of all the types of patterns to which the system will be designed to work. For the character recognition system we need to have English alphabets (both upper case and lower case) and numerical data (0 to 9) as the database. Database usually consists of different fonts in case of printed recognition system or predefined handwritten characters in handwritten character recognition system. The characters are grouped according to their area so that efficiency of the system increases by reducing the effective comparisons.

Chapter 3

CHARACTER RECOGNITION USING CORRELATION

Correlation is a signal-matching technique. It is an important component in digital communication system. It is often used in signal processing for analyzing functions or series of values, such as time domain signals. A correlation is useful because it can indicate a predictive relationship that can be exploited in practice. In this chapter, character recognition using correlation method is explained.

3.1 Correlation

In signal processing correlation can be defined as the technique which provides the relation between any two signals under consideration. The degree of linear relationship between two variables can be represented in terms of a Venn diagram as in figure 4.1. Perfectly overlapping circles would indicate a correlation of 1, and non-overlapping circles would represent a correlation of 0.

For example questions such as "Is X related to Y?", "Does X predict Y?", and "Does X account for Y?" indicate that there is a need for measuring and better understanding of the relationship between two variables. The correlation between any two variables 'A' and 'B' can be denoted by *"R_{AB}"* as shown in figure 4.1. Relationship refers to the similarities present in the two signals. The strength of the relation will always be in the range of 0 and 1. The two signals can be said to be completely correlated if the strength of their relationship is 1 and are completely non-correlated if the strength of the relationship is 0.

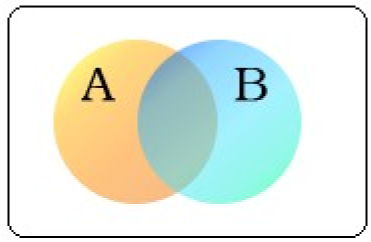


Figure 3.1: Venn diagram representation.

3.1.1 Types of correlation techniques

There are two types of correlation techniques that can be employed in the field of signal processing. They are as follows,

- Linear correlation
- Circular correlation

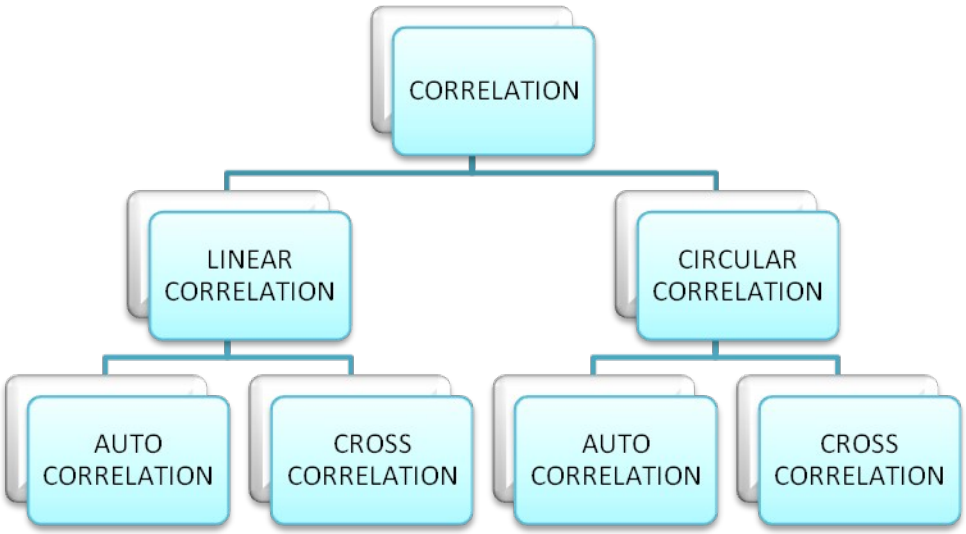


Figure 3.2: Classification of correlation techniques

Also the linear and circular correlations can be further classified into auto-correlation and cross-correlation as shown in figure 3.2. In case of linear correlation, the samples or the signals are shifted linearly i.e. from left to right, whereas in circular correlation, the right most sample is circularly shifted and thus takes the position of the previous left most sample.

3.1.2 Auto correlation

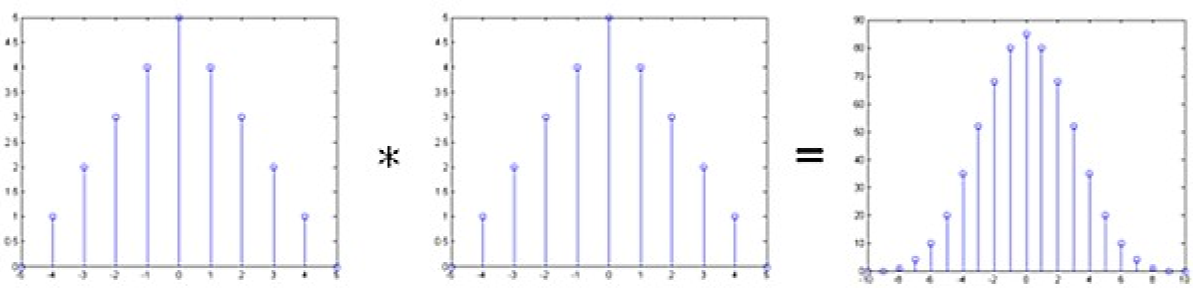
Autocorrelation is the cross-correlation of a signal with itself. It is the similarity between the observations as a function of the time separation between the same signals. It is a mathematical tool for finding repeating patterns, such as the presence of a periodic signal which has been buried under noise, or identifying the missing fundamental frequency in a signal implied by its harmonic frequencies. In an auto correlation, there will always be a peak at a lag of zero, unless the signal is a trivial zero signal.

For continuous function *f*, the auto-correlation is defined as

$$(f * f)(t) = \int_{-\infty}^{\infty} f^{c}(\tau) f(t+\tau) d\tau \qquad (3.1)$$

Where f^{c} denotes the complex conjugate of *f*. Similarly, for discrete functions, the auto correlation is defined as,

$$(f * f)[n] = \sum_{m=-\infty}^{\infty} f^{c}[m]\, f[n+m] \qquad (3.2)$$



(a) (b) (c)

Figure 3.3: (a) and (b) represents x and (c) represents the correlation of x with itself

For example, let x = [1 2 3 4 5 4 3 2 1], then the auto correlation would result in y = [1, 4, 10, 20, 35, 52, 68, 80, 85, 80, 68, 52, 35, 20, 10, 4, 1]. This is shown in figure 3.3. The value at the origin is maximum and hence it can be said that the signal x is correlated with itself by having a relationship value of one.

3.1.3 Cross correlation

In signal processing, cross-correlation is a measure of similarity of two waveforms as a function of time-lag applied to one of them. This is also known as sliding dot product or inner-product. It is commonly used to search a long duration signal for a shorter, known feature. It also has applications in pattern recognition, single particle analysis, electron tomography averaging, cryptanalysis, and neurophysiology.

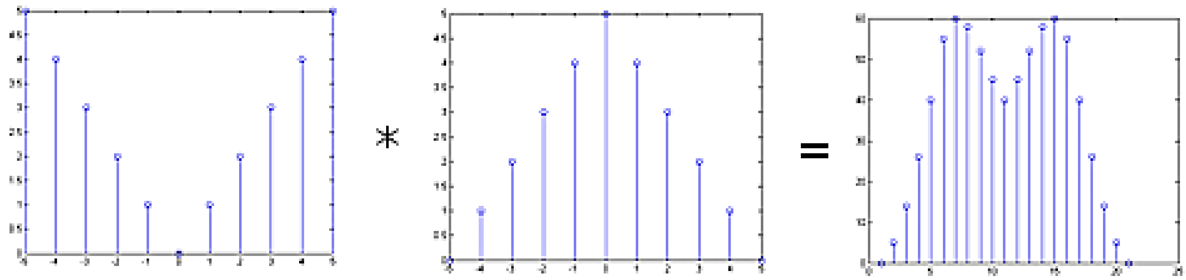
For continuous functions, *f* and *g*, the cross-correlation is defined as,

$$(f * g)(t) = \int_{-\infty}^{\infty} f^{*}(\tau) g(t+\tau) d\tau \qquad (3.3)$$

where *f* * denotes the complex conjugate of *f*. Similarly, for discrete functions, the cross-correlation is defined as

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f^{*}[m]\, g[n+m] \qquad (3.4)$$

For example, let x = [1 2 3 4 5 4 3 2 1] and y = [5 4 3 2 1 0 1 2 3 4 5], the correlation of x and y results in z = [5, 14, 26, 40, 55, 60, 58, 52, 45, 40, 45, 52, 58, 61, 55, 40, 27, 15, 5]. This is shown below in figure 3.4. The cross-correlation is similar in nature to the convolution of two functions. Whereas convolution involves reversing a signal, then shifting it and multiplying by another signal, correlation only involves shifting it and multiplying without reversing.



(a) (b) (c)

Figure 3.3: (a) and (b) represents x & y respectively and (c) represents the correlation of x with y.

If X and Y are two independent random variables with probability distributions f and g, respectively, then the probability distribution of the difference $Y - X$ is given by the cross-correlation $f \star g$. In contrast, the convolution $f * g$ gives the probability distribution of the sum $X + Y$. In probability theory and statistics, the term cross-correlation is also sometimes used to refer to the covariance cov(X, Y) between two random vectors X and Y, in order to distinguish that concept from the "covariance" of a random vector X, which is understood to be the matrix of covariances between the scalar components of X.

3.2 Correlation of 2D signals

As in 1D signals, correlation can also be applied for 2D signals. Since image can be considered as 2D signal with its amplitude being the intensity of a pixel, correlation concepts holds good. The two dimensional correlation is given by,

$$R(j,k,l) = \sum_{n,q,r} x(n,q,r)x(n-j,q-k,r-l) \qquad (3.5)$$

For example, let us consider two images Figure 3.4 shows the result obtained by evaluating correlation of 2D signals. The values shown were obtained by the command,

res = corr2(x,y);

Where 'x' represents image of coins and 'y' represents image of rice. This command returns a value between 0 and 1 in res. The value tells about the strength of the relationship of the images. Thus cross correlation can be implemented for object recognition.

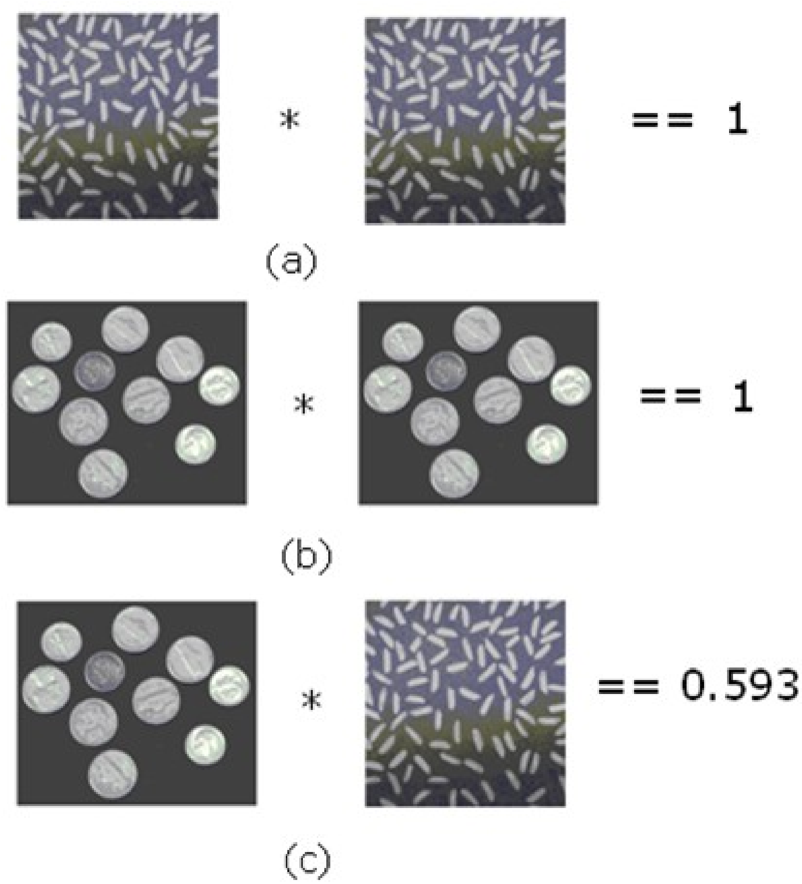


Figure 3.4: (a) auto correlation of rice image, (b) auto correlation of coins image, (c) cross correlation of coins and rice image.

3.3 Implementation of cross correlation for character recognition

In character recognition, each character can be considered as an image and hence 2D-correlation can be implemented for character recognition. Before starting the recognition process, the given document has to go through some preliminary stages where the image can actually be processed so that it can be used to recognize the characters present in it.

Figure 3.5 shows a block diagram of how the character recognition process is carried out through several stages.

The steps involved here are,

1. Image acquisition
2. Preprocessing of the Image
3. Segmentation
4. Character extraction
5. Recognition

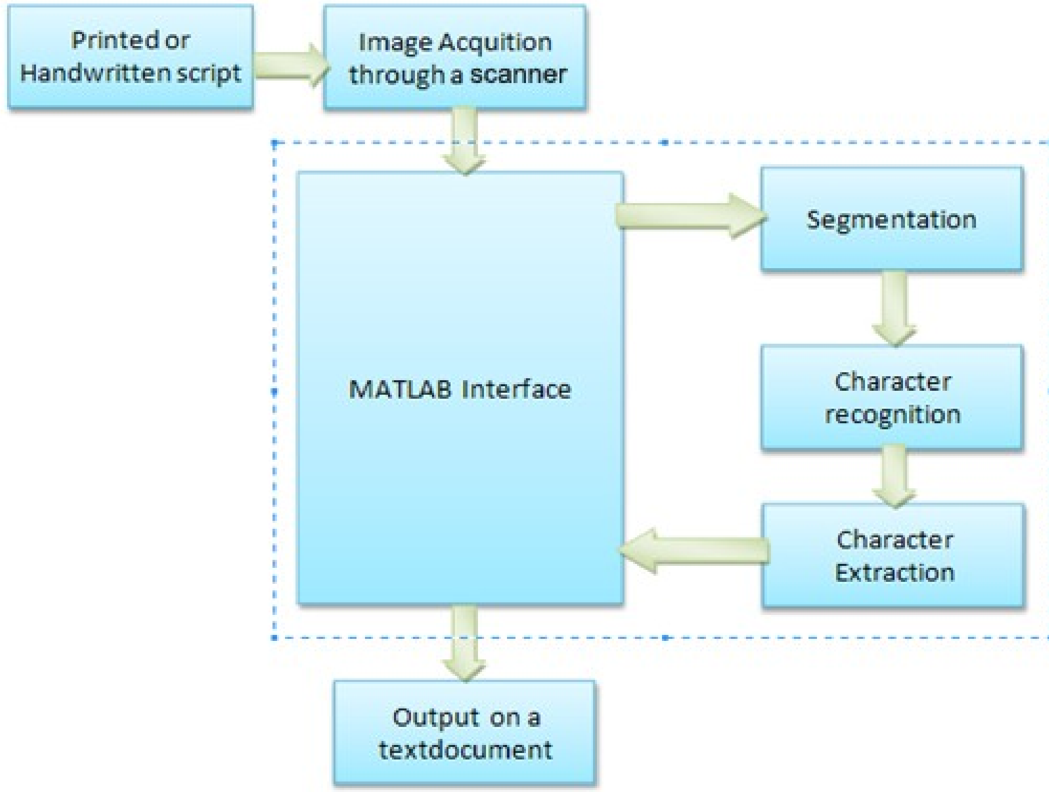
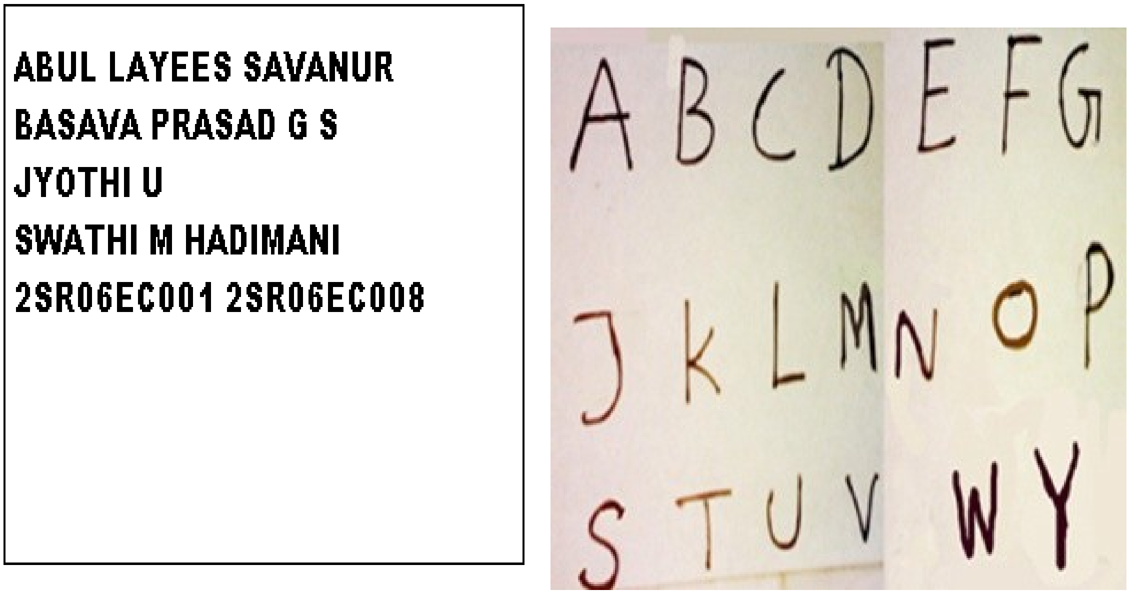


Figure 3.5: Block diagram for implementing recognition process.

3.3.1 Image acquisition

The images are acquired through the scanner. The images are of RGB in nature (Colored). Some of the acquired images are shown in figure 3.6. Figure 3.6(a) is an image of a printed text with the font "*verdana*". Figure 3.6(b) shows a handwritten text[7].



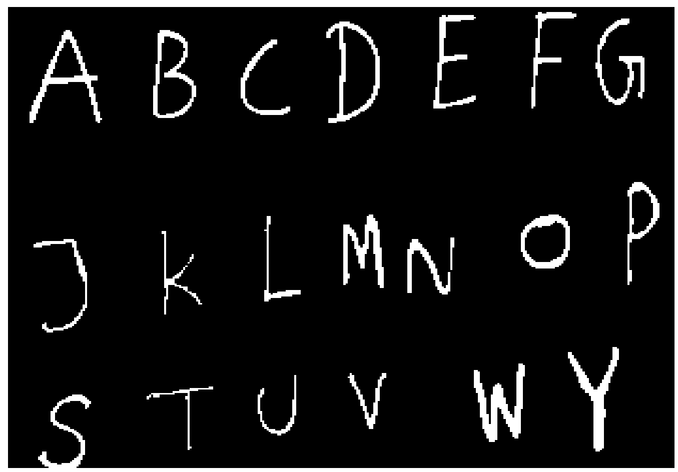
(a) (b)

Figure 3.6: (a) image of printed text, (b) a handwritten image

If carefully observed, one can find some variations in the brightness levels in figure 3.6(a) and some unwanted text printed on the back of the paper in case of figure 3.6(b). These unwanted elements are undesired and thus can be considered as noise. These elements can hinder the performance of the whole system. Thus it is required to remove these noises. Hence preprocessing is required to be carried out on the acquired image.

3.3.2 Preprocessing of the Image

As the captured image is colored in nature, it is required to convert it into a gray image with intensity levels varying from 0 to 255 (8-bit image). Then it is converted into a binary image with suitable threshold (Black=0 & White=1). The advantage is that the handling of the image for further processing becomes easier. This binary image is then inverted i.e. black is made white and white is made black. By doing so, the segmentation process becomes easier [6]. Also some small connected components present in the image is also removed. The preprocessed images are shown in figure 3.7(a) and 3.7(b).



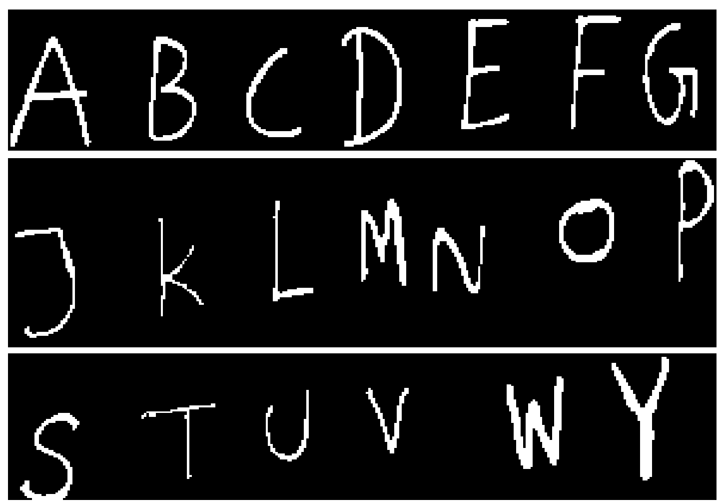
(a)

(b)

Figure 3.7: (a) preprocessed printed text, (b) preprocessed handwritten text

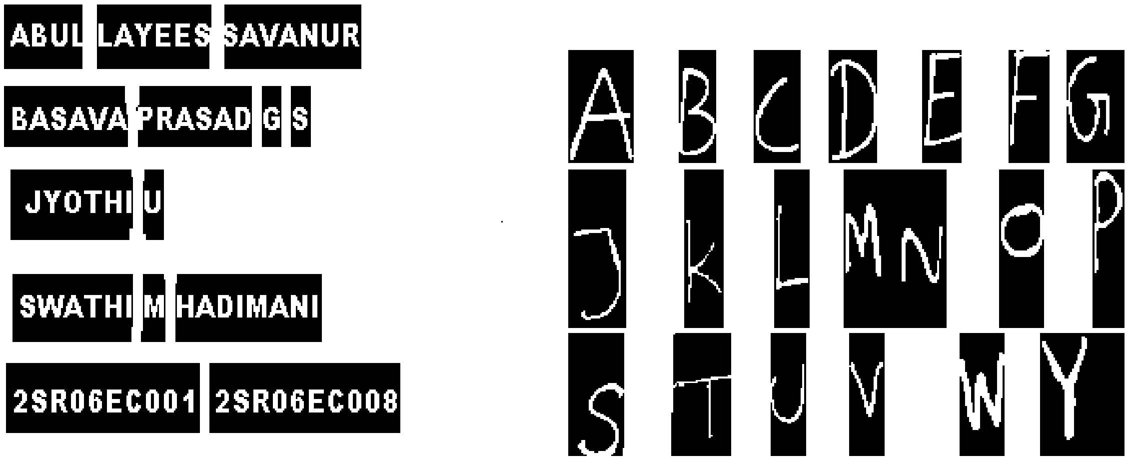
3.3.3 Segmentation

Segmentation is carried out in two stages namely (i) Line segmentation and (ii) Word segmentation. The line segmentation is carried out by scanning the entire row one after the other and taking its sum. Since black is represented by 0 and white by 1, if there is any character present, then the sum would be non zero. Thus the line segmenting is carried out. The lines segmented are shown in figure 3.8(a) and 3.8(b).



(a) (b)

Figure 3.8: (a) Line segmented printed text; (b) Line segmented handwritten text



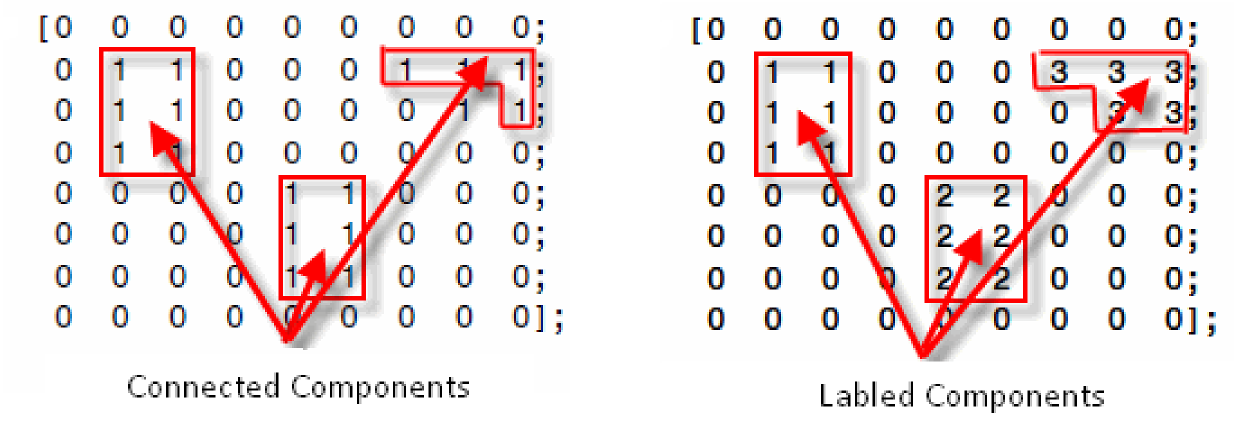
(a) (b)

Figure 3.9: (a) word segmented printed text; (b) word segmented handwritten text

In word segmentation, the same principle used in line segmentation is used. The only difference here is that the scanning process is carried out vertically. The word segmented images are shown in figure 3.9(a) and 3.9(b).

3.3.4 Character Extraction

The characters are extracted through a process called connected component analysis. First the image divided into two regions. They are black and white region. Using 8-connectivity (refer appendix), the characters are labeled. Using these labels, the connected components (characters) are extracted. The extracted characters are then resized to *35 X 25*.



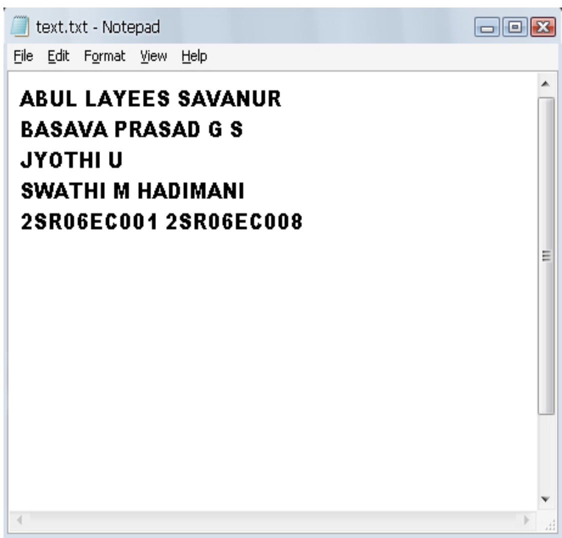
(a) (b)

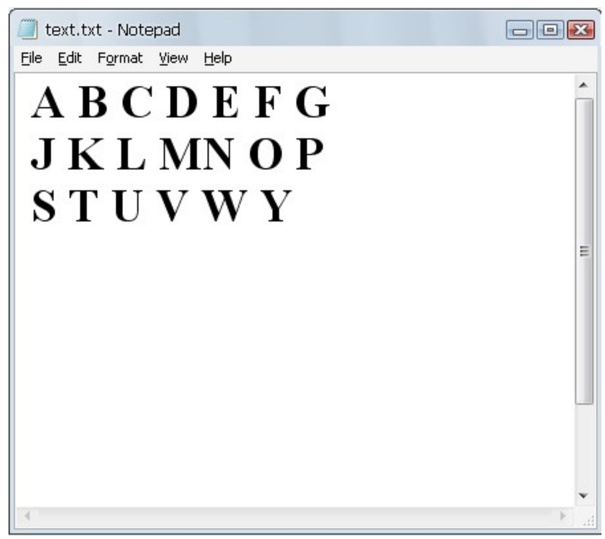
Figure 3.10: (a) connected components in a binary image; (b) labeling of the connected components.

A connected component in a binary image is a set of pixels that form a connected group. For example, the binary image below has three connected components (figure 3.10(a)). Connected component labeling is the process of identifying the connected components in an image and assigning each one a unique label (figure 3.10(b)). The matrix (figure 3.10(b)) is called a *label matrix*. For visualizing connected components, it is useful to construct a label matrix.

3.3.5 Recognition

In the recognition process, each character extracted is correlated with each and every other character present in the database. The database is a predefined set of characters for the fonts Times new roman, Tahoma and Verdana. All the characters in the database are resized to *35 X 25*.





(b)

Figure 3.11: (a) Recognized output for printed text; (b) for handwritten text.

By knowing the maximum correlated value, from the database, the character is identified. Finally, the recognized characters are made to display on a notepad. Figure 3.11 shows the recognized outputs for the segmented images. Recognition for both the formats have some errors but the errors in recognizing a printed text are much lesser than that the errors encountered during the recognition of handwritten characters (figure 3.11(a), figure 3.11(b)).

Chapter 4

ABOUT THE MATLAB SOFTWARE

Software is a tool used for the high speed computation in the computers. Software can be defined as a set of programmes to run an application. There are many types of softwares available that can be used in the field of image processing. The character recognition project is carried out using MATLAB software [1].

MATLAB is a high-performance language for technical computing that integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

Typical uses MATLAB include [1]

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building.

It also includes toolboxes for signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many other areas.

It contains large number of functions. By using some of the functions, the character recognition program using correlation is written.

4.1 GUI using MATLAB

It is a pictorial interface to a program. Good GUI makes programmer easier to use, by providing them with a consistent appearance with control buttons like push buttons, list boxes, and slider menus. By interfacing the program of the interest with inbuilt programs of MATAB, the desired GUI can be constructed [1].

GUI provides an easy way to understand the work, so that the user knows what to expect when he does an action.

Chapter 5

APPLICATIONS OF CHARACTER RECOGNITION

Some of the fields where character recognition can be applied are,

- Text to speech converters:

  By converting handwritten and printed text to editable text format, that can be converted to speech by various available text-to-voice converters that will help the visually impaired persons.

- Postal services:

  To sort out the posts according their destination by reading their address present on their envelope this makes sorting faster and accurate.

- Security applications

  In banking services for reading the *amount, name* of the person present on cheque, DD etc.

- Restoration of old Scripts

  Old scripts can be stored digitally so that they can be reproduced whenever needed.

- The same principle can be used for Number Plate recognition and in the Name Card reader.

- Digital signature verification and much more.

Chapter 6

LIMITATIONS

1. Black colored text printed on the white sheet is preferred for maximum accuracy.

2. No extra light effects must be present while capturing the image.

3. Font style variation must be avoided throughout the text.

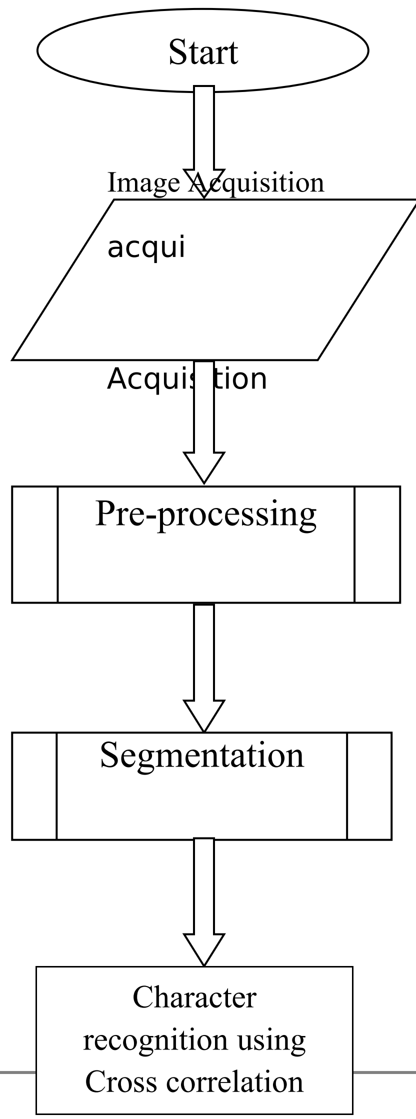
4. It is difficult to differentiate between the characters such as ‘l’&’i’ ; ‘s’&’S’ ; ‘z’& ‘Z’.

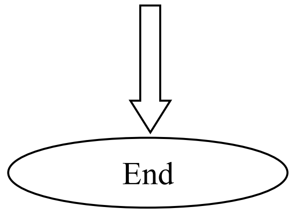
5. Font of text taken as input must match with font of the database images for better accuracy.

Chapter 7

FLOWCHARTS AND ALGORITHM

7.1 Flowchart for Character recognition using Correlation [4]





7.2 Algorithm for Character recognition using Correlation

Step 1 : Start the character recognition process using correlation technique.

Step 2 : Image is captured using camera.

Step 3 : Preprocessing in the captured image is carried out.

Step 4 : Segmentation (Line segmentation and word segmentation) of the preprocessed image is carried out.

Step 5 : Recognition of the characters using correlation technique.

7.3 Image acquisition

Image is captured using a scanner. It may consist of handwritten or printed texts. It is input block of the flowchart.

7.4 Pre-processing

In this pre-processing stage, captured image is inverted, then it is cropped. Now this cropped image is converted into digital image.

7.5 Algorithm for Pre-processing

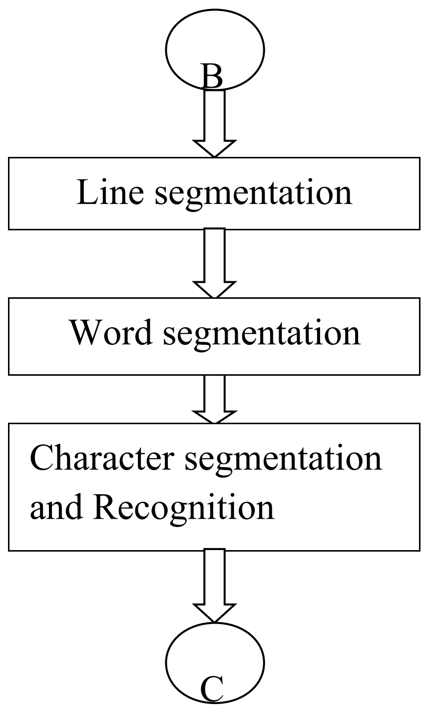
Step 1 : Image is captured using camera, it is input for this stage.

Step 2 : Invert the input image.

Step 3 : Crop the inverted image to the required size.

Step 4 : Convert the cropped image into digital form.

7.6 Segmentation



7.7 Algorithm for Segmentation

Step 1 : Digital image from the pre-processed stage id taken.

Step 2 : Line segmentation is carried out.

Step 3 : Wine segmentation is carried out.

Step 4 :Character segmentation is carried out.

Chapter 8

EXPERIMENTAL RESULTS

The different types of techniques used for character recognition are discussed in the previous chapters. In this chapter the some of the experimental results obtained are shown. Graphical User Interface (GUI) are also showned .

8.1 Results for correlation technique

The results obtained for Character recognition using the correlation technique are discussed**.** For synthetic images, test images and handwritten text, the results obtained are showned in the order.

8.1.1 Image acquisition

The four images captured are shown in the figures 8.1, 8.2 and 8.3. The figures 8.1 is the shows the image of the printed characters (synthetic image). The printed test image is shown in the fig 8.2.These images are further processed according to the algorithm.

8.1.2 Pre-processing

The captured image inverted and it is cropped to the required size. The cropped image is converted into digital form. The pre-processed printed text (synthetic image) is shown in the figures 8.4. The preprocessed printed text (test image) and handwritten text images are shown in the figures 8.5 and8.6 respectively.

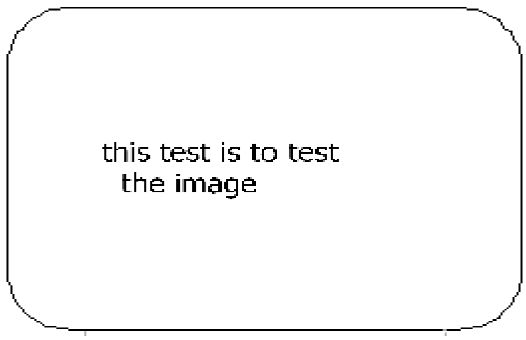


Figure 8.1: Captured image of printed text (synthetic image)

Figure 8.2: Captured image of printed text (Test image)

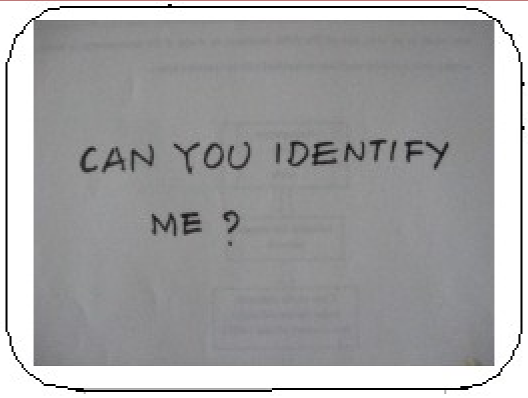


Figure 8.3: Captured image of Handwritten text

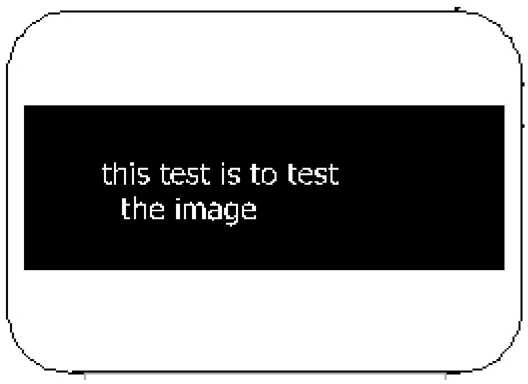


Figure 8.4: Pre-processed image of printed text (synthetic image)

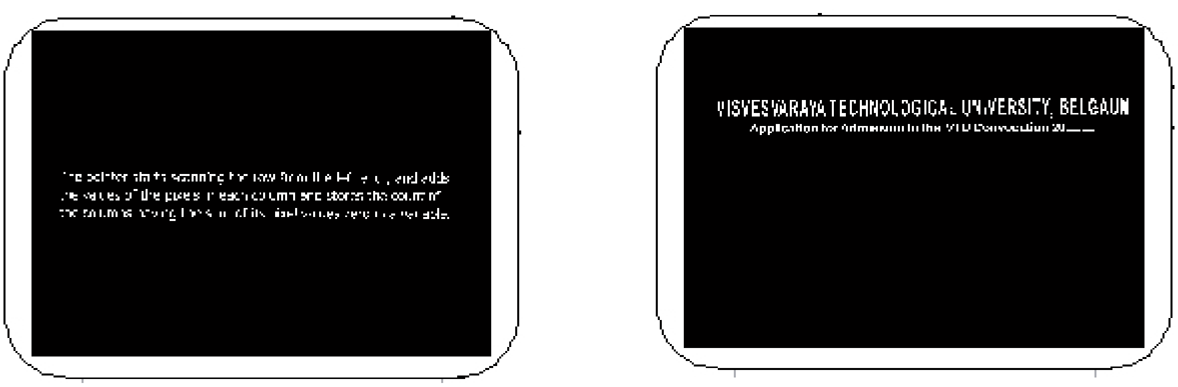


Figure 8.5: Pre-processed image of printed text (test image)

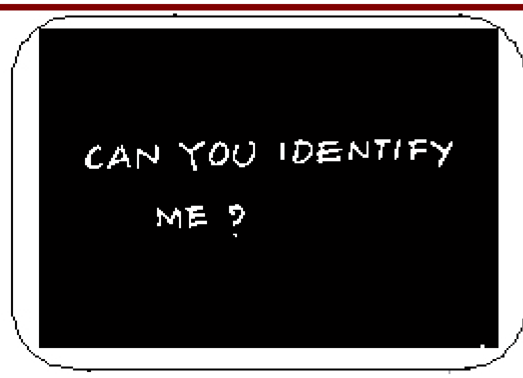


Figure 8.6: Pre-processed image of Handwritten text

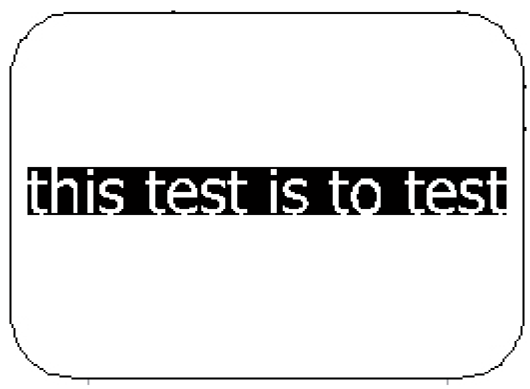


Figure 8.7: Line segmented image of printed text (synthetic image)

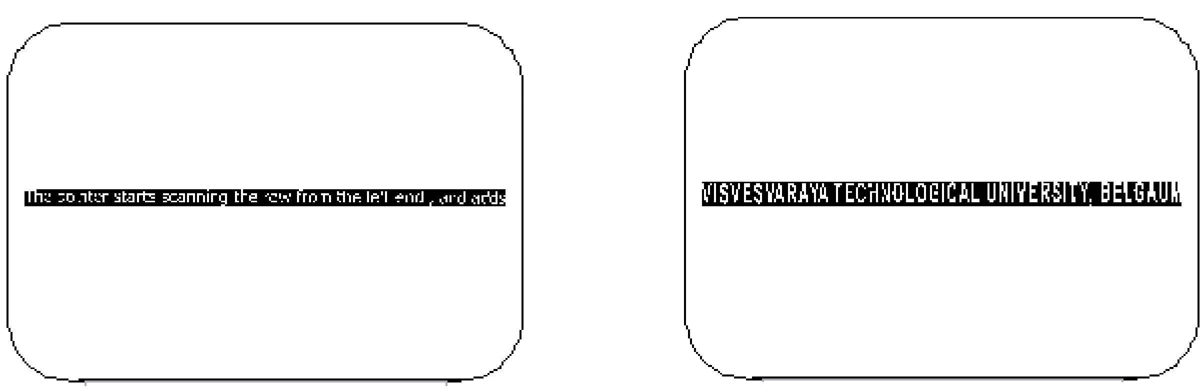


Figure 8.8: Line segmented image of printed text (test image)

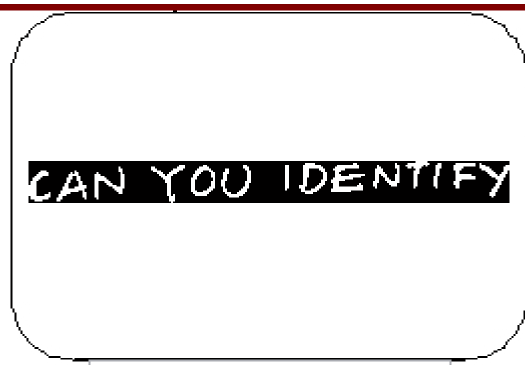


Figure 8.9: Line segmented image of Handwritten text

Figure 8.10: word segmented image of Printed text (Synthetic image)

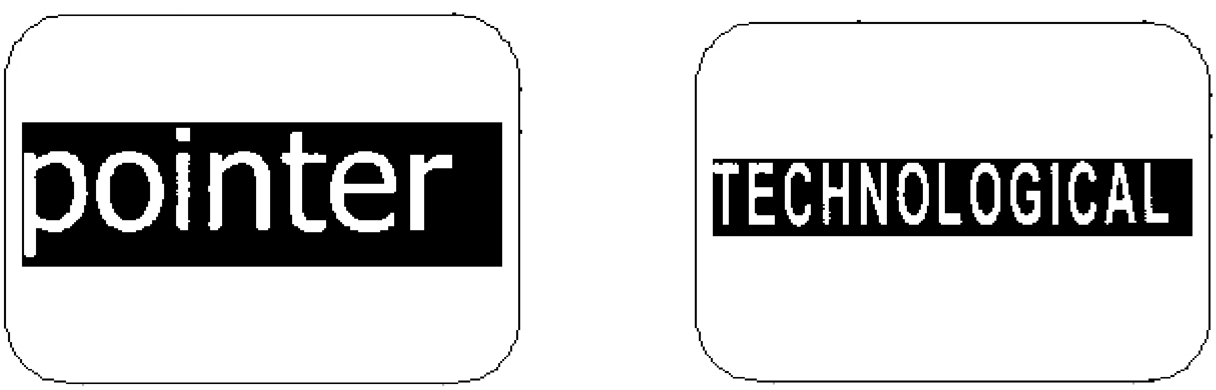


Figure 8.11: Word segmented image of Printed text (test image)

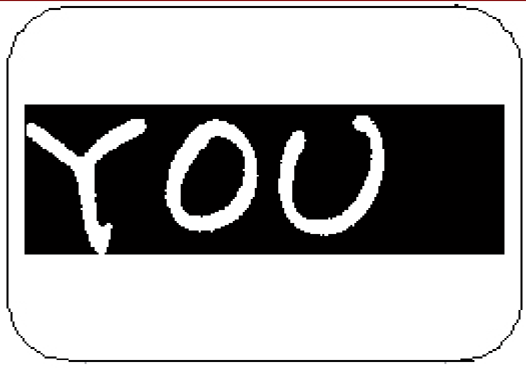


Figure 8.12: Word segmented image of Handwritten text

8.1.3 Line segmentation

The preprocessed images are segmented row-wise (line segmentation). The resulted images of the line segmentation for the figures 8.4, 8.5 and 8.6 are shown in the figures 8.7, 8.8 and 8.9.

8.1.4 Word segmentation

In the line segmented image each word is segmented. The figure 8.10 shows the words segmented from the lines of the figure 8.7 (synthetic image of printed text). The word segmented images for the printed text (test image) and handwritten texts are shown in the figures 8.11 and 8.12.

8.1.5 Character Extraction

The characters extracted from the word in the captured images are shown in the figure 8.13 to 8.16. These each characters are extracted using connected component analysis.

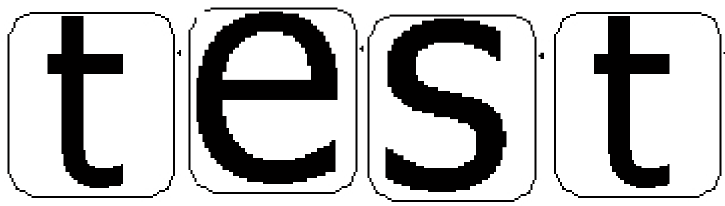


Figure 8.13: Characters extracted from a Printed text

The figure 8.13 shows the characters extracted from the word 'test' in the captured image shown the figure 8.1.

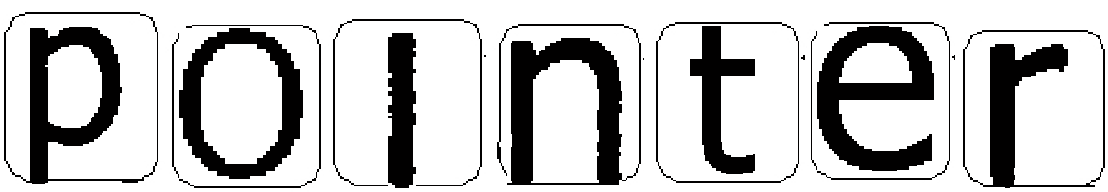
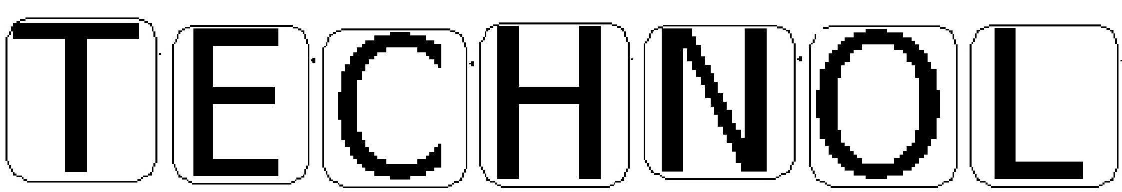


Figure 8.14: Characters extracted from a printed text

The figure 8.14 shows the characters extracted from the word 'pointer' in the captured image shown the figure 8.2.



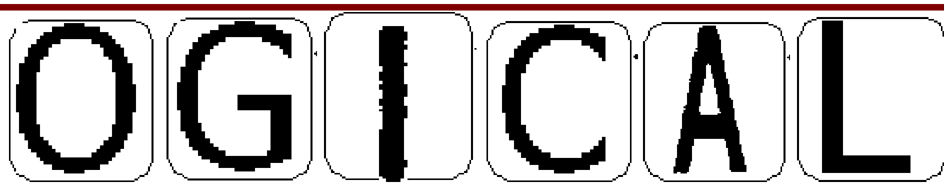


Figure 8.15: Characters extracted from a printed text

The characters extracted from the word ‘technological’ in the captured image shown in the figure 8.2, are shown in the figure 8.15.

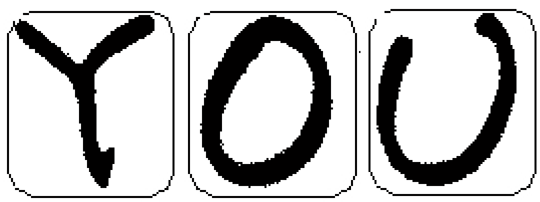


Figure 8.16: Characters extracted from a handwritten text

The figure 8.16 shows the characters extracted from the word ‘YOU’ in the captured image shown the figure 8.3 (Handwritten text).

8.1.6 Notepad output

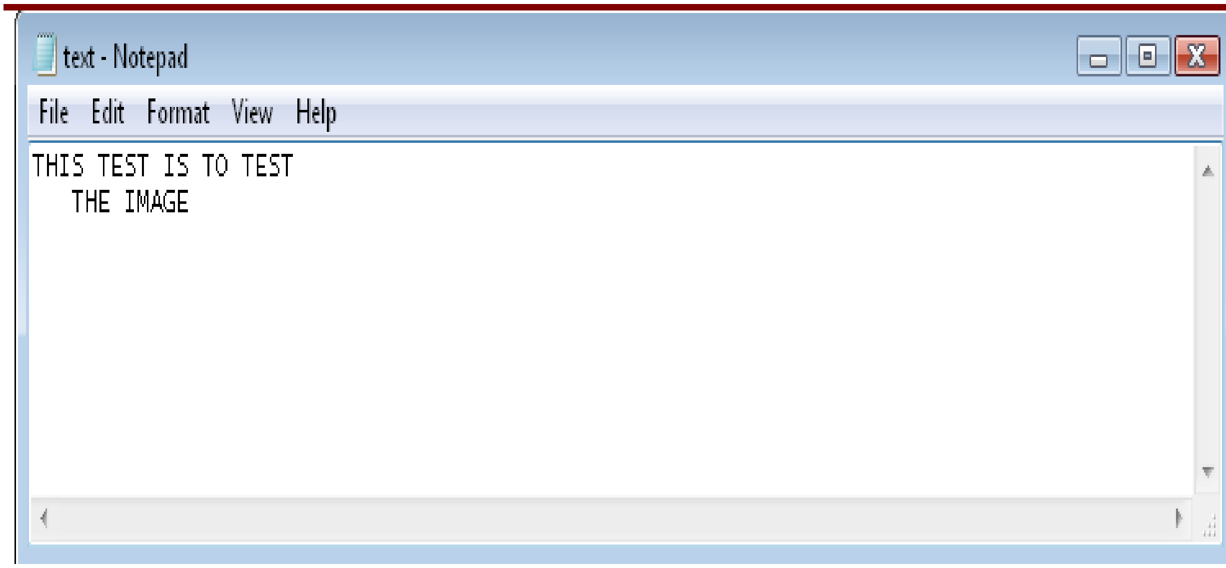


Figure 8.16: Notepad output for printed text (Synthetic image)

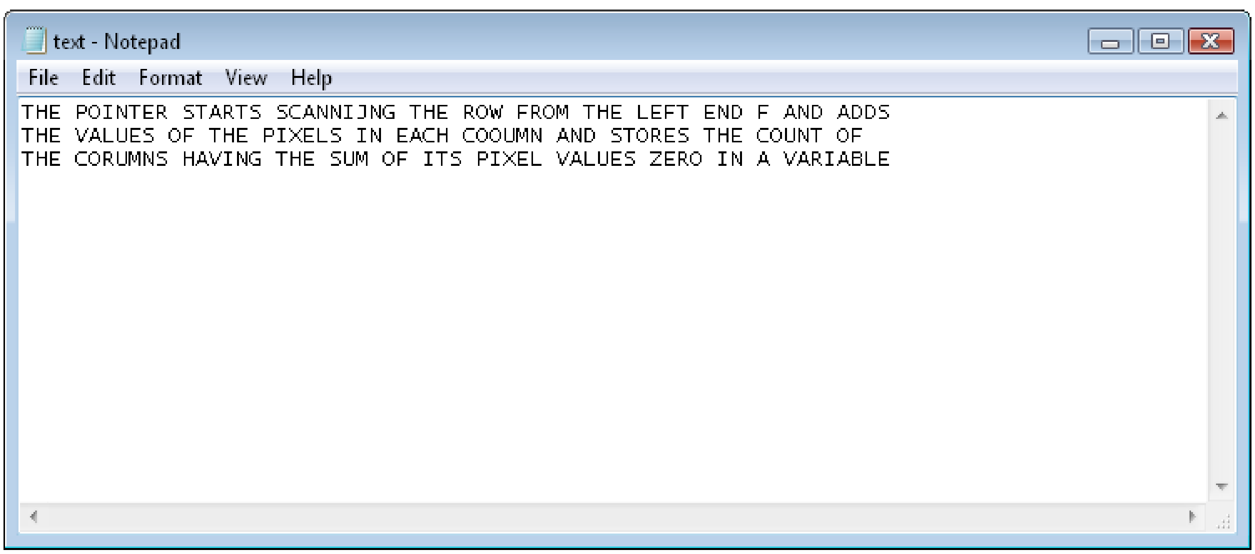


Figure 8.17: Notepad output for Printed text (test image)

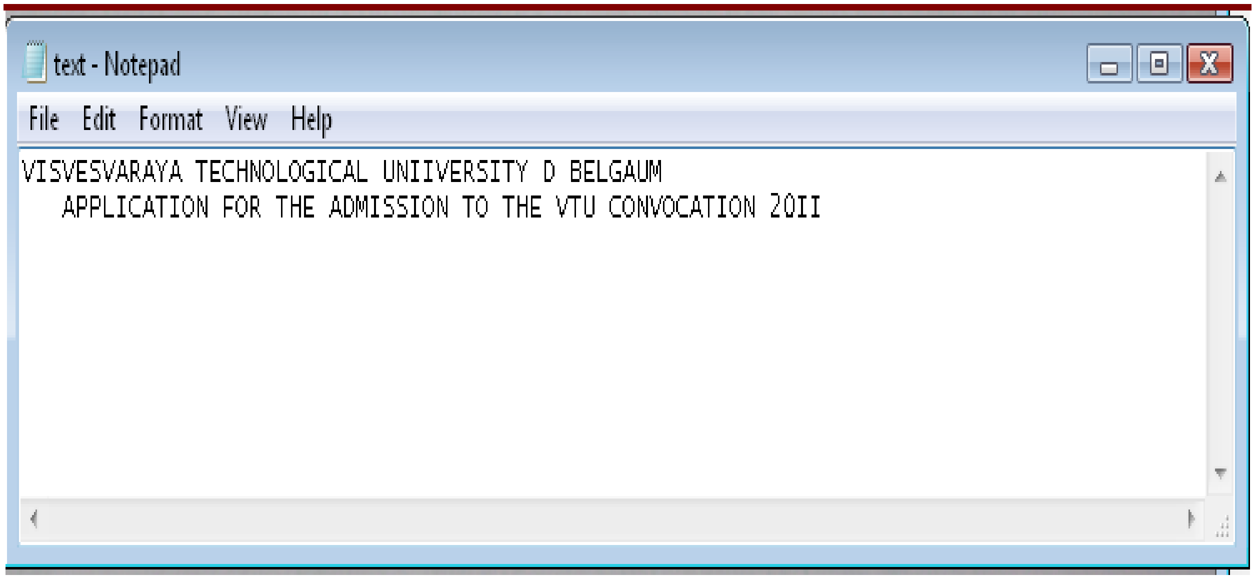


Figure 8.18: Notepad output for Printed text (test image)

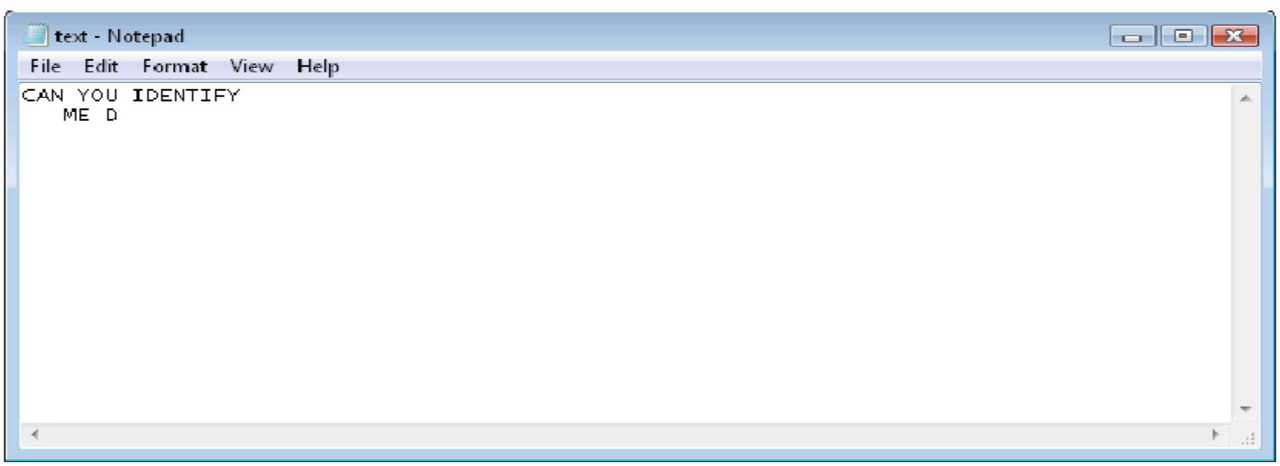


Figure 8.19: Notepad output for Handwritten text

Figures 8.16, 8.17, 8.18, and 8.19 shows the recognized characters obtained in the notepad which is the final output [8].

8.2 GUI

The GUI (Graphical User Interface) figure for the character recognition using the correlation is shown in the figure 8.20. In this figure, all the stages of the character recognition are shown.

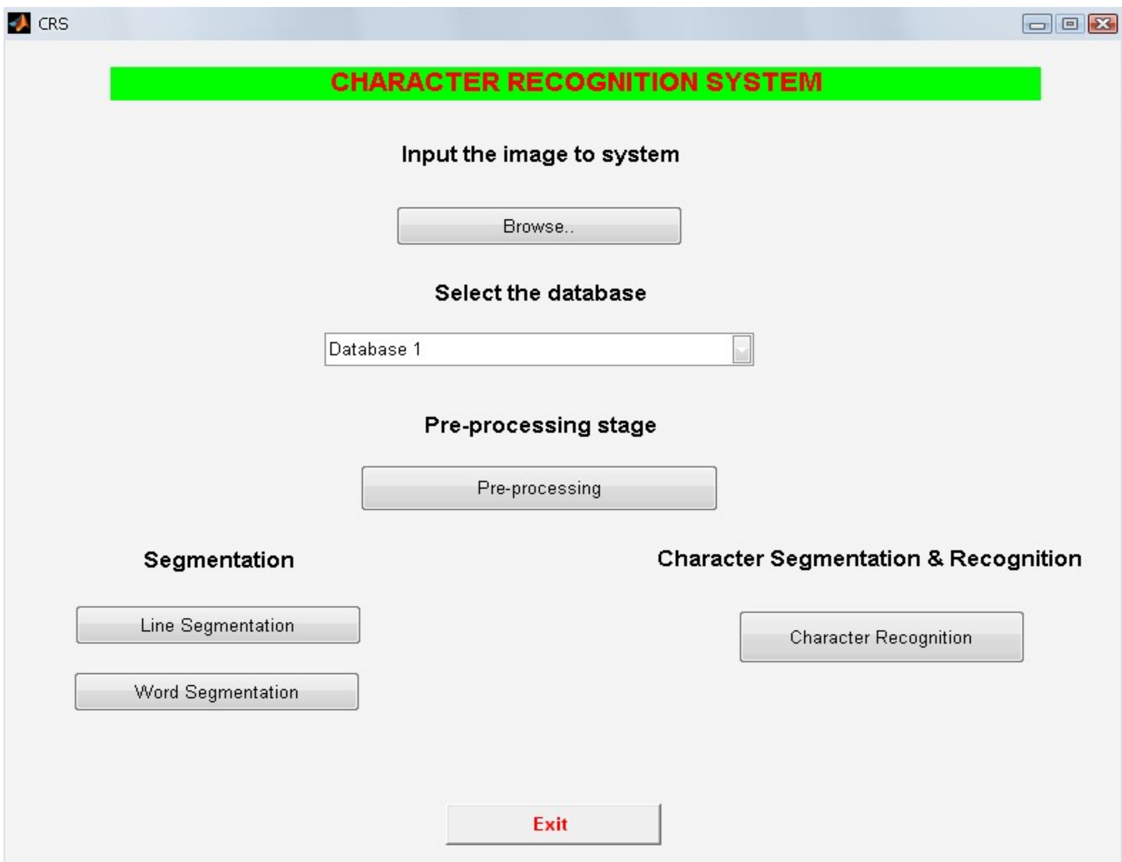


Figure 8.20: GUI for character recognition system.

Chapter 9

CONCLUSION

"Character Recognition" using correlation technique is easy to implement. Since this algorithm is based on simple correlation with the database, the time of evaluation is very less. Also the database which was partitioned based on the areas of the characters made it more efficient. Thus, this algorithm provides an overall performance in both speed and accuracy.

"Character Recognition" using correlation, works effectively for certain fonts of English printed characters. This has applications such as in license plate recognition system, text to speech converters, postal departments etc. It also works for discrete handwritten run-on characters which has wide applications such as in postal services, in offices such as bank, sales-tax, railway, embassy, etc.

Since "Character Recognition" deals with offline process, it requires some time to compute the results and hence it is not real time. Also if the handwritten characters are connected then some errors will be introduced during the recognition process. Hence the future work includes this to be implemented for an online system. Also this has to be modified so that it works for both discrete and continuous handwritten characters simultaneously.

REFERENCES

[1] Rafeal C.Gonzalez, Richard E.Woods, “Digital Image Processing”, third edition 2009.

[2]R. Plamondon, S. N. Srihari, “On-line and off-line handwritten recognition: a comprehensive survey”, IEEE Transactions on PAMI, Vol. 22(1), pp. 63–84, 2000.

[3] Negi, C. Bhagvati and B. Krishna, “An OCR system for Telugu”, in the Proceedings of the Sixth International Conference on Document Processing, pp.1110-1114, 2001 .

[4] Jayarathna, Bandara, “A Junction Based Segmentation Algorithm for Offline Handwritten Connected Character Segmentation”, IEEE Nov. 28 2006-Dec. 1 2006, 147 – 147.

[5] Dr.-Ing. Igor Tchouchenkov, Prof. Dr.-Ing. Heinz Wörn, “Optical Character Recognition Using Optimisation Algorithms”, Proceedings of the 9th International Workshop on Computer Science and Information Technologies CSIT’2007, Ufa, Russia, 2007.

[6] John Makhoul, Thad Starnert, Richard Schwartz, and George Chou, “On-Line Cursive Handwriting Recognition Using Hidden Markov Models and Statistical Grammars”, IEEE 2007

[7] Sonka, Halvac, Boyle, “Digital image processing and computer vision”, first Indian reprint 2008, page 345-349.

[8] Michael Hogan, John W Shipman, “OCR (Optical Character Recognition): Converting paper documents to text”, thesis submitted to New Mexico Tech Computer Center, 01-02-2008

[9] Robert Howard Kasse, “A Comparision of approaches to online handwritten character recognition”, submitted to the department of EE&CS for the degree of Ph.D at MIT, 2005.