Faculty of Electrical and Electronic Engineering Technology

OPTICAL CHARACTER RECOGNITION (OCR) ON IMAGES USING TEMPLATE-MATCHING AND IMAGE CORRELATION

NURUL HASYA BINTI ROJEHAN

Bachelor of Computer Engineering Technology (Computer Systems) with Honours

2022

OPTICAL CHARACTER RECOGNITION (OCR) ON IMAGES USING TEMPLATE-MATCHING AND IMAGE CORRELATION

NURUL HASYA BINTI ROJEHAN

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Computer Engineering Technology (Computer Systems) with Honours

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2022

DECLARATION

I declare that this project report entitled "Optical Character Recognition on Images using Template-Matching and Image Correlation" is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	[illegible]
Student Name	:	NURUL HASYA BINTI ROJEHAN
Date	:	7/1/2022

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Engineering Technology (Computer Systems) with Honours.

Signature	:	[illegible]
Supervisor Name	:	TS. DR. ROSTAM AFFENDI BIN HAMZAH
Date	:	7/1/2022

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEDICATION

Alhamdulillah, praise to the Almighty Allah SWT.
This thesis is dedicated to:
My mother,
Mrs Rozilah Binti Harun

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRACT

Optical Character Recognition (OCR) has grown in popularity as a result of its widespread use in converting photos into editable machine-coded text in the multimedia and digital fields. Optical Character Recognition (OCR) is a method for extracting content from a digital image by increasing its quality. Pre-processing, Segmentation, Feature Extraction, and Classification are all part of the OCR system. To identify the font and retrieve the text, template matching and correlation were utilised. The goal of this project is to extract text from photos that contain critical information, with the text serving as the output.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRAK

Sepanjang tahun, Pengesahan Huruf Optik (OCR) telah menjadi satu permintaan yang tinggi disebabkan penggunaanya dalam menukarkan imej kepada huruf komputer yang boleh diubah dalam bidang digital dan multimedia. Teknik Pengesahan Huruf Optik (OCR) adalah proses untuk mendapat teks dengan memperbaiki kualiti imej digital. Sistem OCR terdiri daripada Pre-pemprosesan, Pengekstrakan ciri, Pengelasan dan Pemprosesan Pasca. Untuk pemprosesan pasca, padanan templat dan korelasi digunakan untuk mengenalpasti huruf komputer dan mengekstrak teks tersebut. Tujuan projek ini ialah untuk mendapat teks dari gambar yang mempunyai maklumat yang sangat penting di mana teks tersebut akan keluar melalui buku nota.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ACKNOWLEDGEMENTS

Predominantly, I am inclined to express my gratitude to my supervisor, Ts. Dr Rostam Affendi Bin Hamzah for his precious admonishment, words of insight and tolerance all over this project.

I am also honour-bound to Universiti Teknikal Malaysia Melaka (UTeM) for the monetary support which validates me to achieve the project. Not slipping my colleagues for their enthusiasm for sharing their thoughts and opinions regarding the project.

My highest admiration goes to my parents, and family members for their fondness and prayer during the span of my study. Without their support, I would not have been qualified to complete my final year project.

Finally, I would like to thank everyone who is involved in this project either directly or indirectly, associates and classmates, the faculty members, as well as other solitaries who are not listed here for being cooperative and obliging.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

TABLE OF CONTENTS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UTeM
اونيۈرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF TABLES

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF FIGURES

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF SYMBOLS

$\sum x$	- Total of x

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF ABBREVIATIONS

A	-	The template grey level image
$\bar{A}$	-	The average grey level in the template image
B	-	The source image section
$\bar{B}$	-	The average grey level in the source image

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF APPENDICES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 1

INTRODUCTION

1.1 Background

The term OCR is an acronym for Optical Character Recognition. It is a technique for detecting text within a digital image. Text recognition in scanned documents and photographs is a typical application. OCR software can turn a physical paper document or an image into a text-searchable electronic form. OCR, which is implemented in MATLAB software, is a widely used programme that is well-known for its many features in the image processing toolkit. OCR is a text recognition technology that uses photos or videos to recognise text in a multimedia document. Many studies and developments have been conducted on OCR to produce a precise text extraction result. Many approaches are used in the development of OCR, including image processing, pre-processing, and other approaches that are introduced and integrated into the OCR system. All of these strategies have their own set of advantages and disadvantages when it comes to text extraction. In addition, the OCR is subjected to a variety of methods and algorithms to obtain the desired output.

This project intends to create an OCR system in MATLAB programme to extract the text using an OCR system. MATLAB was chosen because it is simple to implement, has free software access for students, and is simple to understand the process in OCR. MATLAB stands for MATrixLABoratory, a programme that converts images into matrixes so that subsequent processes can be performed quickly. As a result, the colours, edges, intensity, texture, and pattern in the image can be easily recognised and identified. Pre-processing, segmentation, feature extraction, and classification are the four main components of an OCR system. The images will be pre-processed, resulting in images that are smoother than the originals. Each character is converted from the segmented word. The feature extraction will then take the character's information and compare it to the existing template for classification.

1.2 Problem Statement

In today's technological world, advanced multimedia technology has become a source of concern for daily living, as it can be confusing for some people. People like to take pictures, but it is risky to maintain them because there could be problems with them. Despite this, an architecture for an OCR system was developed to address the issue. However, to share the image with others, the text information included within the image must be typed manually. The processes inside the OCR system can assist users in quickly sharing information in text format by converting digital photos into extractable text. However, there could be an issue if the text were extracted incorrectly. To avoid this, the OCR system's performance must be evaluated using a variety of photos to ensure that the text extraction result is accurate.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.3 Project Objective

These are the three goals outlined below:

a) To present a template-matching and picture correlation-based framework for OCR.

b) To match the fonts on an image to the proposed project.

c) To evaluate the performance of an OCR system with various types of fonts found in photos.

1.4 Scope of Project

The approach of extracting text from photos is focused on a specific group of people with specific needs. First, it focuses on individuals who want to avoid keeping information within photos, as well as others who may harm device storage utilisation. Next, this strategy focuses on a wide community, particularly in multimedia and digital, to make the process of exchanging information via social media or a website more efficient.

اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

For extracting text from photos, optical character recognition is now widely used. Many studies turned out to be conducted to boost and enhance the current system's production. This chapter examines earlier optical character recognition research papers for the past five years.

2.2 Overview

Optical Character Recognition (OCR) has become well-known, and its application in extracting text from photos has enticed individuals to adopt it, particularly those with a passion for multimedia. People are increasingly using smartphones to capture photographs to save the information contained in the image as technology advances.

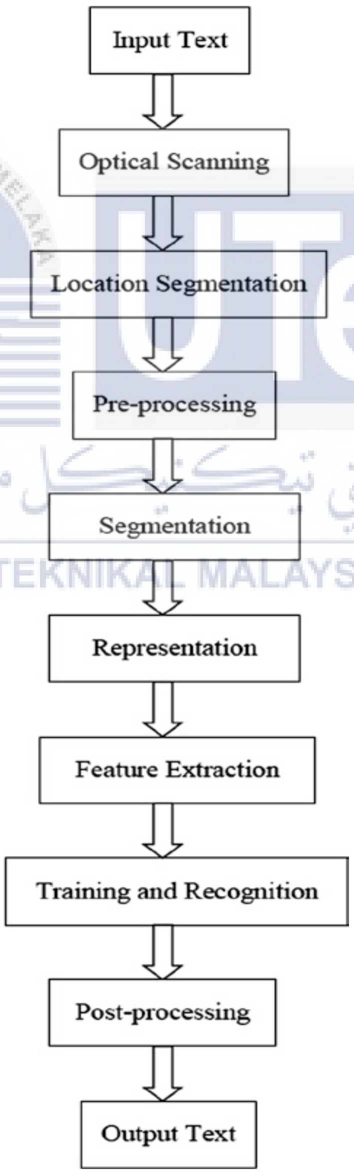
In general, OCR is a type of data entry method that enters data into a system in alphabetic, numeric, or symbolic form. Because OCR is widely employed in the digital world, demand for OCR has continued to rise till now. People learn the simplest technique to save data by using OCR, which converts images into text, which can then be copied and simply forwarded to others.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3 Related Research

2.3.1 Optical Character Recognition Systems

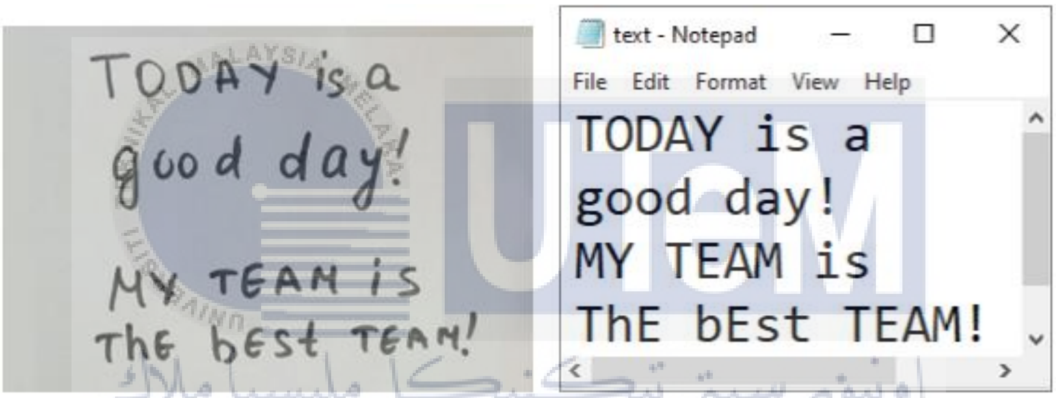
This study described the procedures used by the existing OCR system in detail, as well as the source and remedy to the problem. A typical OCR system is made up of various parts [1]. After the images have been printed, this system will be turned off. Handwritten and printed characters can be accomplished; however, the quality of the scanned image must be considered. Figure 2.1 shows the elements of an OCR system.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.1 Components of OCR System

The OCR system, which is based on MATLAB (MATrixLABoratory), is well-known for transforming images into matrices and allowing numerous processes to be applied to the images to obtain the desired result. The images' colours, intensity, edges, texture, and pattern may all be detected using this software [2]. The scanner is used as a medium to capture images into the computer by using the OCR process. During the segmentation phase, the text region in the image will be recognised, and the symbol will be extracted [3]. To go on to the next stage, the recognised symbol will go through a series of steps that include pre-processing and noise removal. Each symbol was identified by comparing the extracted symbol classes from the preceding procedure.

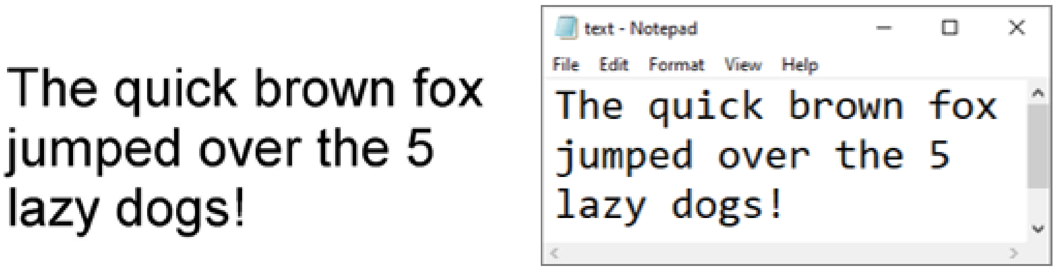


TODAY is a
good day!
MY TEAM is
The best TEAM!

TODAY is a
good day!
MY TEAM is
ThE bEst TEAM!

اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.2 Handwritten Sample and Output



The quick brown fox
jumped over the 5
lazy dogs!

The quick brown fox
jumped over the 5
lazy dogs!

Figure 2.3 Text Image Sample and Output

To evaluate the outcome, the system was reviewed based on three characteristics throughout the process. Character classification is proportionately proportioned in terms of recognition rate. There are some characters that the system does not recognise, which contributes to the rejection rate. The fail character, on the other hand, is highlighted by the OCR system, making it easier to spot for manual repair. In terms of error rate, the system will not notice a misclassified character, but the error can be detected and corrected manually. Even though the OCR system is cost-effective, the time spent detecting and correcting the OCR system's errors is more essential [4]. There will be only a 1% error indicated in a 99 per cent correct identification rate.

2.3.2 Steps Involved in Text Recognition in OCR

In this study, various steps are reviewed including text recognition, classification of manuscript OCR systems according to text type, and application-oriented recent OCR research. Reducing noise, contrast, enhancement, and image sharpening are examples of low-level image processing operations [5]. Pre-processing is a crucial step before moving on to feature extraction since it ensures that the results are appropriate for the subsequent phases. Binary or grey pictures are used in the majority of OCR applications. Without conducting the pre-processing stage, the photos may have watermarks or a non-uniform backdrop, making recognition harder.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The filters are used to cancel out the image's high or low frequency. Smoothing is the process of removing high frequencies from an image while boosting or edge detection is the process of removing low frequencies [6]. The following Figure 2.4 displays the primary image and 2.5 displays the image that has been applied with Prewitt and Canny edge detection methods.

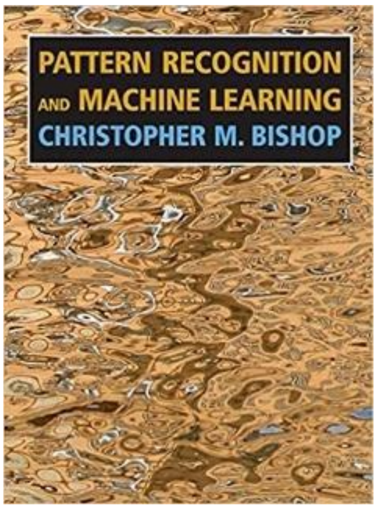
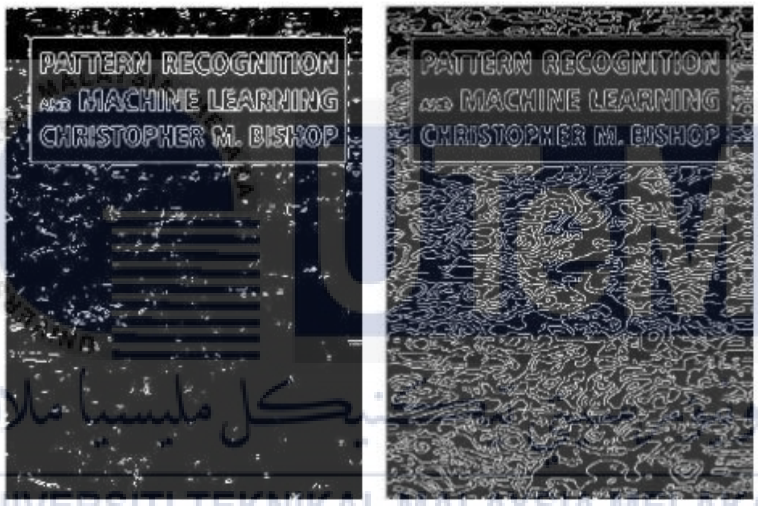


Figure 2.4 Original Image



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.5 Prewitt Method and Canny Method

Mask and point processing are two types of spatial image filtering processing. Global thresholding is one of the steps in spatial image filtering processing. Image thresholding is a technique for isolating information from its surroundings. As a result, this method is classified as global and local thresholding and is commonly used to grey-level or scan colour images. The global thresholding method selects a value for the entire image from the intensity histogram. A grey-level image is automatically reduced to a binary image via global thresholding. The contained adaptive thresholding approach employs different values for each pixel based on information from the surrounding area. Figure 2.6 displays the global threshold appealed using Otsu’s method.

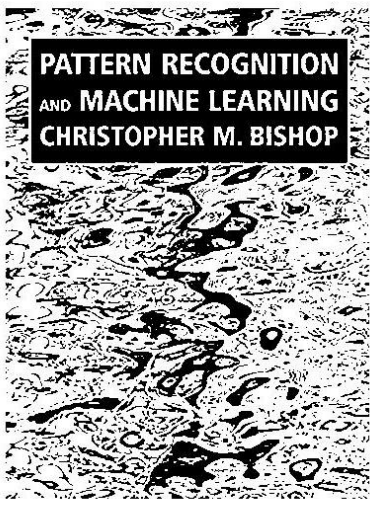
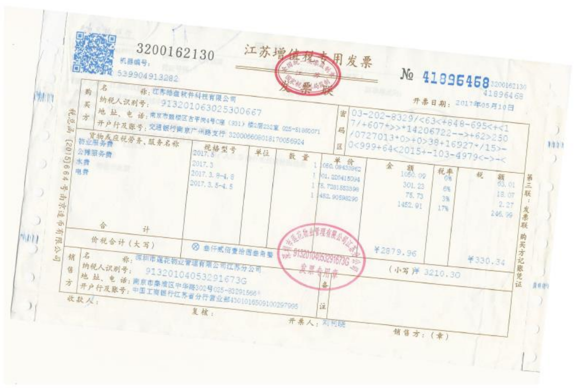


Figure 2.6 Otsu's Method

2.3.3 Template Matching-based Method for Intelligent Invoice Information Identification

This work presents a technique for intelligently identifying invoice details based on template matching, which involves picture pre-processing, template matching, optical character recognition, and information exporting to extract the relevant information [7]. To automatically detect invoice information, the method suggested in this research uses image processing approaches and presents a template matching-based procedure [8]. The invoice image is first pre-processed to remove the effects of rotation, inclination, and backdrop, and then matched to extract the essential information region [9]. The characters in the region are identified using OCR technology, and the invoice information identification results from different regions are then transported to an excel sheet in the same style.

UTeM UNIVERSITI TEKNIKAL MALAYSIA MELAKA



3200162130 江苏增值税专用发票 № 41896468

机器编号: 539904913282

发票联

开票日期: 2017年05月10日

购买方	名　　称: 江苏拾盘软件科技有限公司 纳税人识别号: 91320106302530066 7 地 址、电 话: 南京市鼓楼区古平岗4号C座 (931) 楼2层232室 025-51860071 开户行及账号: 交通银行南京广州路支行 320006606018170056924	密码区: 03-202-8329/<63<+848-695<+<1 7/+607+>>+14206722-->+62>250 /0727013+0>+0>38+16927+/15>- 0<999+64<2015+-103-4979<->-<

货物或应税劳务、服务名称	规格型号	单位	数量	单价	金额	税率	税额
物业服务费	2017.5		1	1050.09433962	1050.09	6%	63.01
公摊服务费	2017.3		1	301.226415094	301.23	6%	18.07
水费	2017.3.8-4.8		1	75.7281553398	75.73	3%	2.27
电费	2017.3.5-4.5		1	1452.90598290	1452.91	17%	246.99
合　计					¥2879.96		¥330.34
价税合计 (大写)	⊗ 叁仟贰佰壹拾圆叁角整				(小写) ¥3210.30		

销售方	名　　称: 深圳市莲花物业管理有限公司江苏分公司 纳税人识别号: 91320104053291673G 地 址、电 话: 南京市秦淮区中华路302号025-83291566 开户行及账号: 中国工商银行江苏省分行营业部4301016509100297995	备注

收款人:　　复核:　　开票人: 刘利陵　　销售方: (章)

第三联: 发票联 购买方记账凭证

税总函 [2015]664号南京通币有限公司

91320104053291673G 发票专用章

Figure 2.7 Scanned Invoice Image

Figure 2.7 shows the original image of the invoice utilised. Because of its irregularity, this sort of skimmed image cannot be detected straight, necessitating pre-processing. Thus, secondary rotation is needed. Contour extraction is the underlying principle of secondary rotation. Contour extraction can extract the appropriate image location information for the desired contour [9]. The position and ratio of the secondary rotation must be established before it can be carried out. Each point's rotation angle in the portray abides constantly when rotating due to the relative invariance of position. As a result, the point of rotation is placed to the image's intermediate to simplify the rotation phase.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

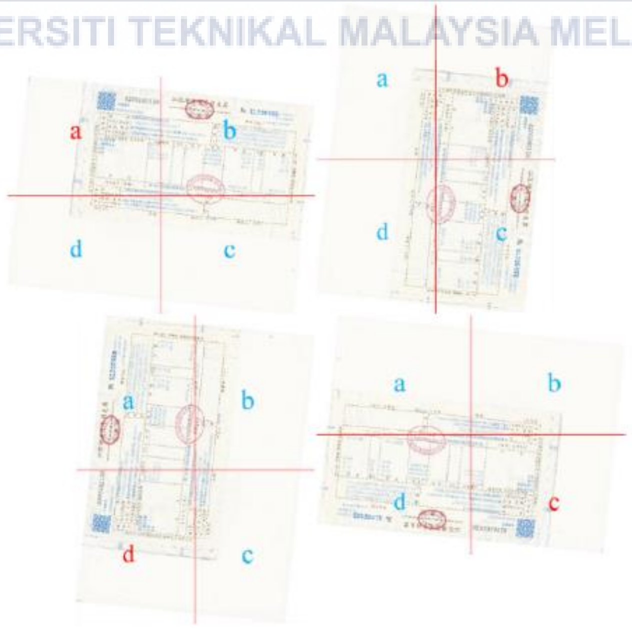
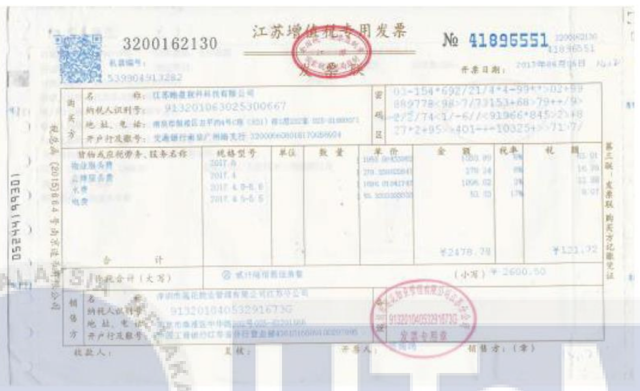


Figure 2.8 Possibilities for First Rotation

The changes in rotation angle as the QR code latitude changes, as seen in Figure 2.8. The image of the invoice is separated into four latitudes: a, b, c, and d, with the red letter indicating which zone the QR code corresponds to in each image. The image will remain motionless if the QR code is given a score of at the area a. The image must be rotated in the other direction at 90°, 180°, and 270° degrees in the remaining three cases of b, c, and d, respectively. Figure 2.9 shows the image after competing for two rotations.


3200162130 江苏增值税专用发票 № 41896551

机器编号: 539904913282

开票日期: 2017年[illegible]

购买方 名称: [illegible]
纳税人识别号: 913201063025300667
地址、电话: [illegible]
开户行及账号: [illegible]

密码区 03-154*692/21/4*4-99*[illegible]
889778<98>7/73153+68>79++/9>
[illegible]/74<1/-6//<91966*845>2[illegible]
27*2+95[illegible]401-+-10325+>>71-7/

货物或应税劳务、服务名称	规格型号	单位	数量	单价	金额	税率	税额
[illegible]	2017.6			[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	2017.4			[illegible]	[illegible]	[illegible]	[illegible]
水费	[illegible]			[illegible]	[illegible]	[illegible]	[illegible]
电费	[illegible]			[illegible]	[illegible]	17%	[illegible]
合计					¥2478.78		¥121.72

价税合计（大写） [illegible] （小写）¥2600.50

销售方 名称: [illegible]
纳税人识别号: 91320104053291673G
地址、电话: [illegible]
开户行及账号: [illegible]

收款人: 复核: 开票人: 销售方: (章)

Figure 2.9 Image after Secondary Rotation

2.3.4 A Study on Optical Character Recognition Techniques

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The authors of this study resolved and investigated OCR's hypothetical and numerical models. For the recognition of patterns or alphabets, the techniques of optical character identification or classification (OCR) and magnetic character recognition (MCR) are commonly used. In general, the alphabet can be found in a range of pixel images, and they can be scribbled or stamped in any series, form, or orientation [10]. Alternatively, in MCR, the alphabets are printed using magnetic ink, and the studying machine classifies them based on the unique magnetic field that each alphabet creates. MCR and OCR are both used in banking and other commercial applications.

Input and target data make up data. An input and output vector could be used to guide a single alphabet. The resolution of the information will be determined by the length of the input vector. If they worked concerning the fourth network, the range would be 77 elements, with each line following the previous one. The values of the items could be 0 or 1. The vector element's value is 1 if the grid component of the vector-representation image constituent covers more than half of the letter's content characteristics, otherwise, it is 0.

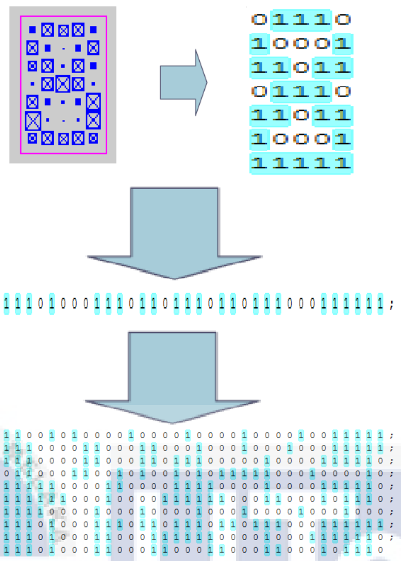


Figure 2.10 Input Data

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

We get the vector by testing this for entire grid components. We have ten numbers in the basic neural network. We will have ten rows of these types of vectors this time, with each line acting for a number. This is how the matrix of inputs is described. This type of statistic should be used as inputs to the neural network in MATLAB. The output matrix with NxN dimension matrix filled with zeros will be generated for a network having N outputs with ones along the main axis. This part of the recognition process was done by hand.

Alphabet recognition systems can bring benefits by supplying an interface that facilitates interfaces between man and machine. Archiving documents, automatic substantiation of checks, data entry, and a wide range of business-oriented applications are just a few of the request areas where OCR plays a critical role. The entire optical character identification approach is estimated to work with bi-tonal pictures of black passage in white surroundings. However, due to differences in picture spatial colour, programming approaches, resolutions, and compression formats, there are a variety of image configurations in this ostensibly simple imaging domain. Typical bi-tonal reproduction scanned document images have spatial resolutions of 200 dots per inch. For high-quality OCR, this resolution may not always be sufficient. With papers scanned at 300 dpi or higher, modern character recognition engines perform better. Online libraries that want to practically protect the superiority of the original article typically use greyscale or smooth fonts.

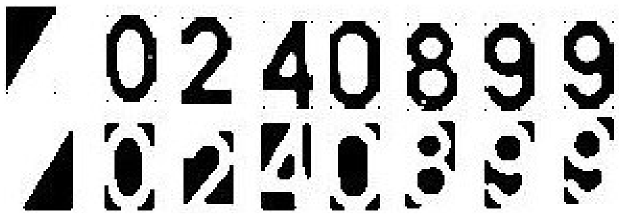


Figure 2.11 Segmented Characters

Segmented Characters is made up of the following characters:

a) The colour of the plate is being changed.

b) The contrast is being increased.

c) Remove all objects in the alphabets that have fewer pixels than predicted.

d) Noise is removed from the obtained items.

e) Isolating each letter of the alphabet.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3.5 Optical Character Recognition using Template-matching and Back Propagation Algorithm

This study discusses the scanned images that are converted mechanically or electronically, text that contains graphics, camera-captured images, images scanned, and as well as picture recognition with smeared or broken characters. OCR is a desktop programme that was created with Java IDE and MySQL as the database [11]. The rate of character recognition is directly proportional to the image quality like picture resolution. Because of the many possible differences in backdrop and fonts, scanned photos are more challenging.

The software is written entirely in Java. The first image has been loaded into the starting module, from whence it is sent in its first state to the pixel extraction module. The image's dimensions can have different sizes. We used various approaches to remove undesirable images, noise, and undesirable text from the image during pre-processing. Cropping, resizing, rotating the image left or right, zoomed in or out as well as the image's resolution are all options for the adjusted images. If undesired text or graphics are eliminated from the image, the excess picture noise in the example substantially decreased, increasing the accuracy of text recognition.

Coloured pictures are frequently found in camera photos. The image must be grayscale for image processing. Grayscale images consist of three colours: red (r), green (g), and blue (b) [12]. The only information in a greyscale image is intensity. This image is also referred to as black and white since it is made up entirely of grey tones with black accents is the enervated and white being the most powerful. The light's intensity in a single group of the spectrum of electromagnetic waves is measured in greyscale photographs. Pixel intensity has been represented as a range of values ranging from 0 to 1, with fractional values in between. Instead of a file, an image will be accepted as an image buffered in Java.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

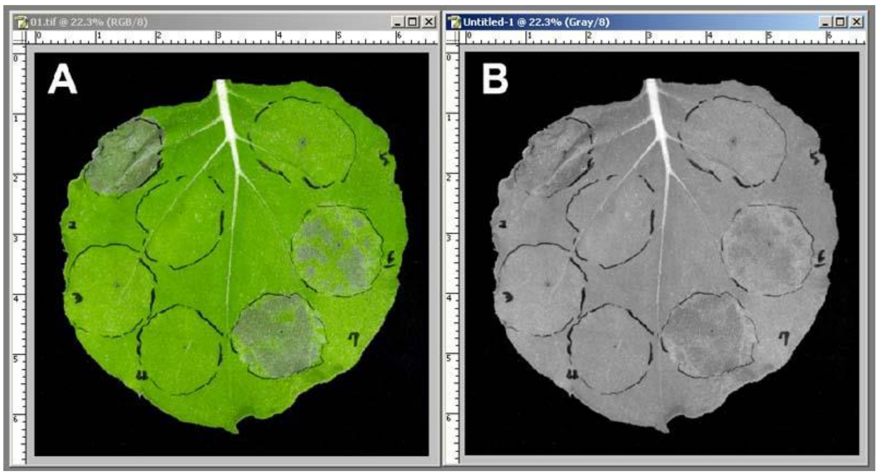


Figure 2.12 Grey Scaling Conversion of Image

An algorithm for supervised learning is back-propagation. Back-propagation is sometimes well known as "error back-propagation." It is based on error correction and minimises the error by spreading it backwards. It operates in two stages.

Pass 1:

a) In forward propagation, the training data is used to the network's nodes. The propagation at the output nodes must be generated.

b) Finally, we have outputs generated by the network's reaction. The network's weight is fixed during forwarding propagation.

Pass 2:

a) In backward propagation, synaptic weights are modified concerning the error correction rule.

b) To generate an error, we subtract the desired response from the actual response.

c) The estimated error is passed backwards through the network to change synaptic weights, and the process is repeated until the desired output is obtained.

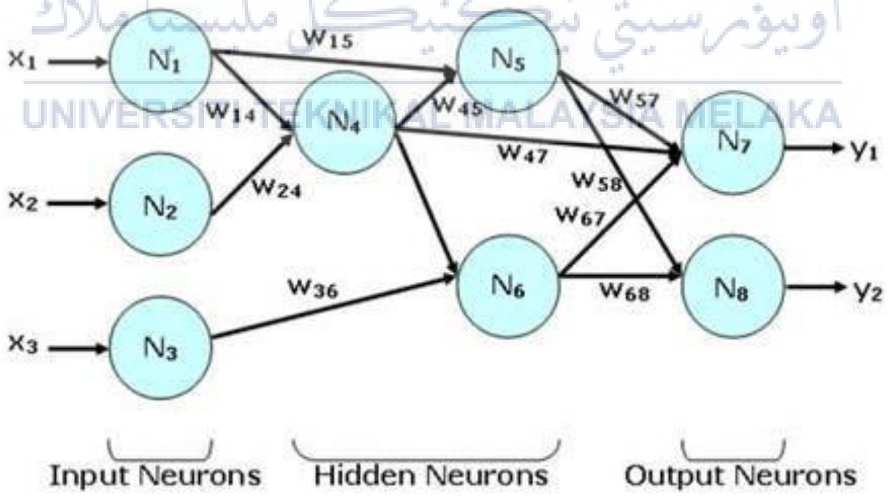


Figure 2.13 Layers of Back-propagation

2.3.6 Consumer Service Number Recognition Using Template Matching Algorithm for Improvements in OCR based Energy Consumption Billing

The images captured by camera-equipped Smartphones and other portable devices have gotten a lot of notice in the area of computer vision in recent years. These images may provide useful information on the image's content. Understanding the information in situations is critical for both humans and computers. The ability to recognise text from these scenes is essential . In a variety of applications, OCR has been used to speed up data entering [13]. For electricity consumption billing, smart metering needs additional infrastructure and constant broadband connectivity. For developing countries, improving existing infrastructure by investing in new resources is a difficult undertaking. The author expands on his previous work on the energy consumption billing process in this paper.

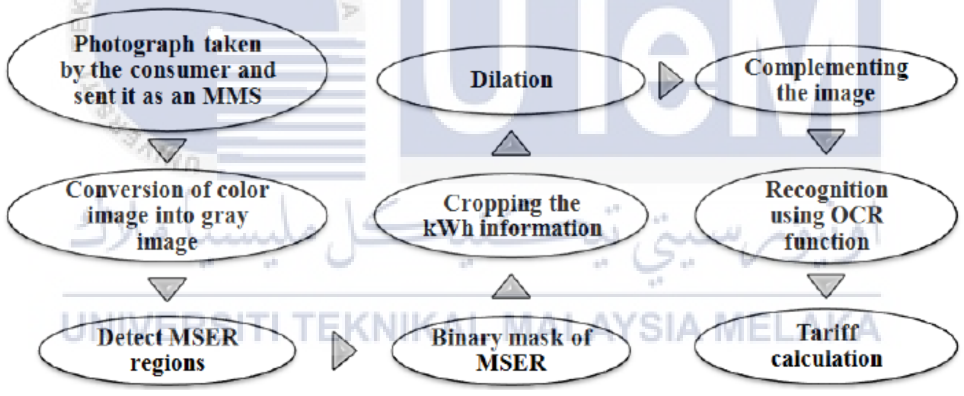

Photograph taken by the consumer and sent it as an MMS
Conversion of color image into gray image
Detect MSER regions
Binary mask of MSER
Cropping the kWh information
Dilation
Complementing the image
Recognition using OCR function
Tariff calculation

Figure 2.14 OCR based Electricity Billing Process

The text recognition process of a consumer service number from a picture of an energy metre display is described in the algorithm below.

The steps involved in the suggested approach are as follows.

i) Load the digital image of the energy metre display.

ii) Crop the region where the consumer service number is identified.

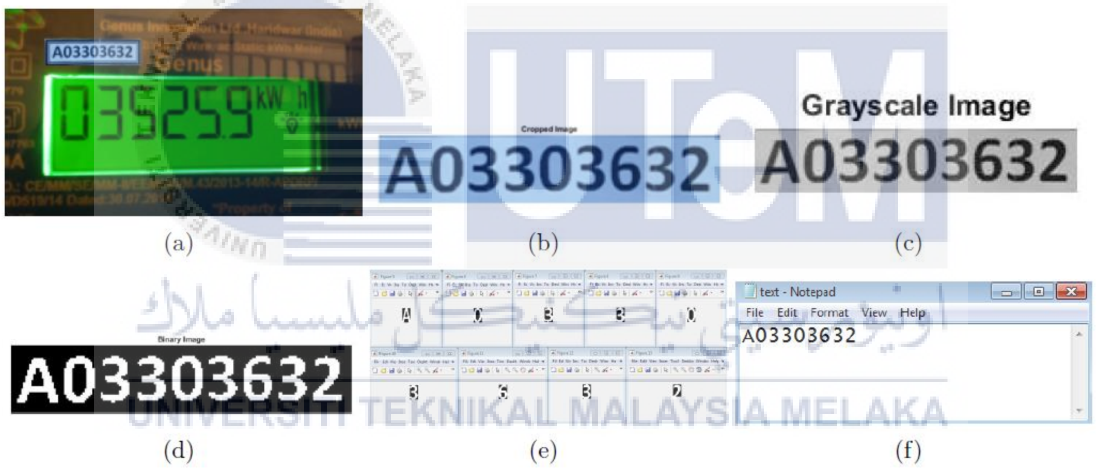
iii) Prepare the cropped image for post-production.

iv) Using the CCA approach, segment the individual characters from the digital image.

v) Using the template matching method, recognise the text that is present [14].

vi) To save the energy usage value, make the service number a variable.

The recognised word comprises the consumer's service number (prefixed with the letter 'A'), and it can be used as a variable to store the kWh data obtained through the MLOCR approach. The energy metre display with the consumer service number is shown in Figure 2.15 (a). The image should be sent to the utility server via MMS. The customer service number is the printed number that is pasted on the energy metre display. The printed consumer number will be clipped initially, as a glimpse in Figure 2.15 (b).



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.15 Step in Text Recognition

As opposed to 1 bit bi-tonal black and white photos, the cropped image is transformed to greyscale. The image has been exhibited in greyscale (c). Figure (d) depicts the binary picture created by converting a greyscale image to binary. Binary images are sometimes known as bi-level or two-level images. The binary image has only two possible values for each pixel: black and white. It means that each pixel is represented by a single bit, with a value of either 0 or 1. Figure (e) depicts the binary image's segmented characters. The process of segmenting a word into isolated characters to perform template matching is known as character segmentation. The size of each isolated character is reduced to 42 x 24 pixels. The isolated characters of the customer service number are correlated with the reference templates provided. As seen in, the identified characters will be saved in a text document (f). The kWh information will be saved using the recognised service number

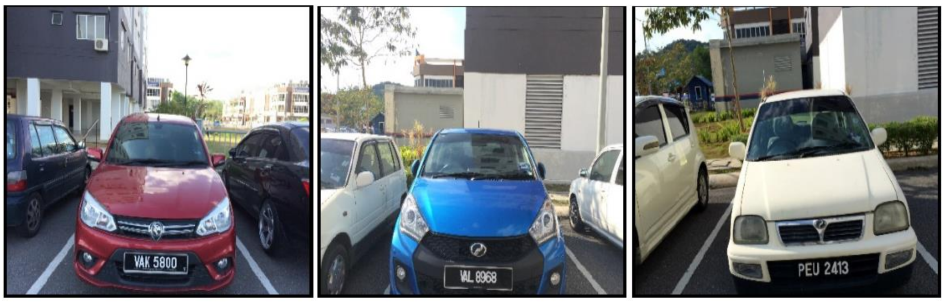
2.3.7 Automatic Vehicle Licence Plate Recognition System based on Image Processing and Template Matching Approach

An automobile licence plate recognition mechanism is a critical skill that may be used to identify engine vehicles anywhere on the planet. Admission to the building, safety, parking authority, road vehicles management, and pace control are only a few of the applications. However, the system is only able to detect the licence number, and the acquired data must be controlled by an operator [15]. As a result, this study provides an image processing and template matching methodology for an autonomous licence plate recognition system. This project entails creating a simulation programme that can recognise licence plate characters using a taken image of a car as input [16].

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The segmented number plate will next be recognised using a combination of approaches for image processing and an OCR method [17]. Colour transformation technique, image division utilising Otsu's thresholding, noise reduction, subtraction of images, picture clipping, and the bounding box characteristic are among the image processing techniques. To investigate the written disposition on the segmented licence plate photo and provide product data make up of characters, an OCR builds on template-matching methodology is applied.

The authors listed several image processing algorithms that can be used to generate a segmented region of interest in licence plate photos as well as recognising the characters on the plate.



(a) Car A (b) Car B (c) Car C

Figure 2.16 The Original Car Images

It could be tough to separate the licence plate because the segmentation technique requires deliberation of the three R, G, and B values, which the original car photos are in RGB format. To make the picture segmentation process easier, it is always simpler to transform an RGB image to a grayscale image, later than to a binary image. Otsu's thresholding was used to separate the characters of the licence plate from the background region throughout this operation. As illustrated in Figure 2.17, the output of Otsu's thresholding is a binary image.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

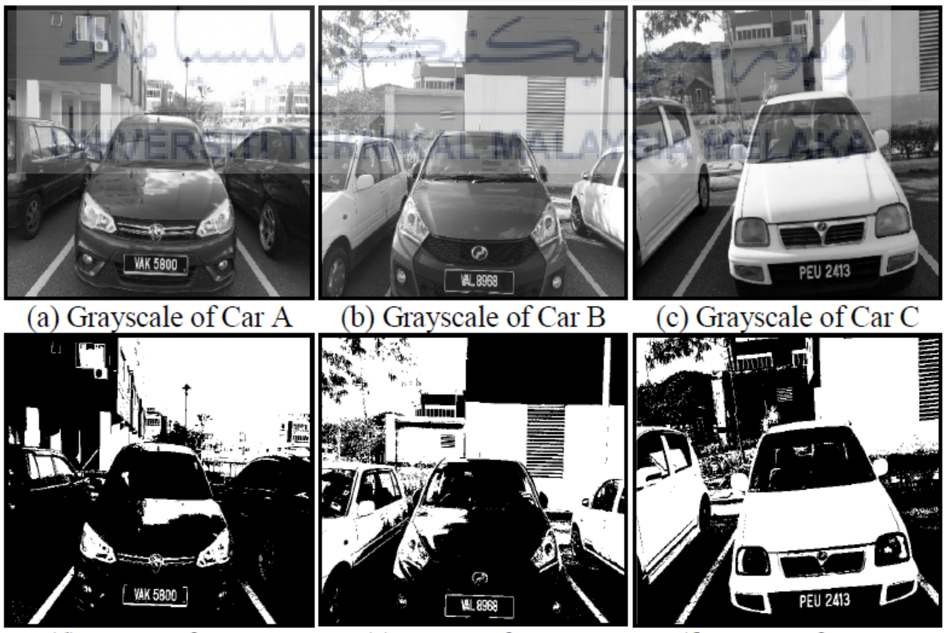
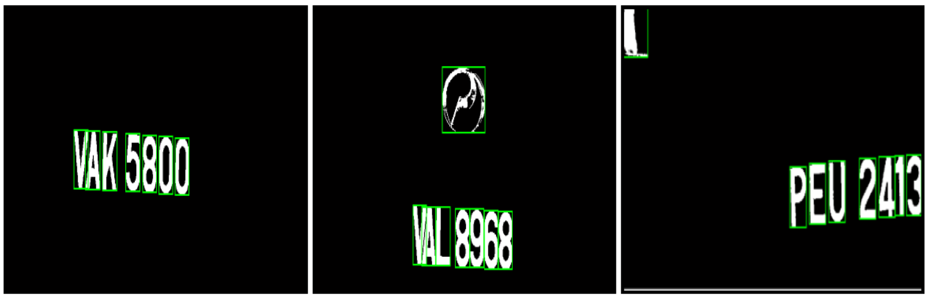


Figure 2.17 The Car Images after Undergoing Grayscale Conversion and Otsu's Thresholding Techniques

Following that, an image cropping procedure was used cropping the segmented licence plate image utilising standardised values. This procedure is carried out to guarantee that only the output image contains licence plate characters, hence improving the accuracy of the character recognition operation. The bounding box has been generated for every one of the split characters licence plates to succeed in the character recognition operation.

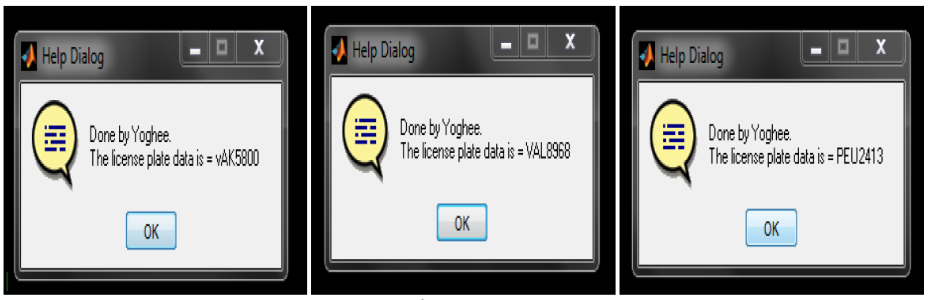


(a) Car A (b) Car B (c) Car C

Figure 2.18 Results of Images of Segmented Licence Plate after Performing the Image Cropping and Bounding Box Process

Finally, a character recognition procedure based on template matching is utilised to recognise information reproduced on an image and provide data output including characters. Figures 2.18 and 2.19 illustrate the outcomes of making use of the bounding box feature and OCR using the template-matching methodology to the three distinct automobiles, respectively.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Done by Yoghee.
The license plate data is = vAK5800

Done by Yoghee.
The license plate data is = VAL8968

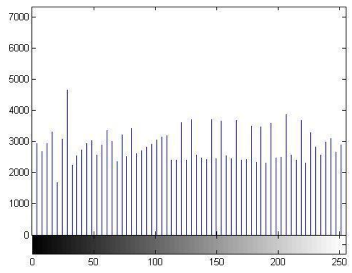
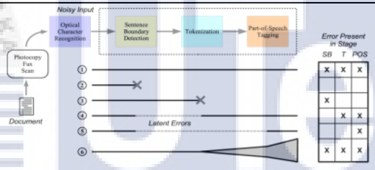
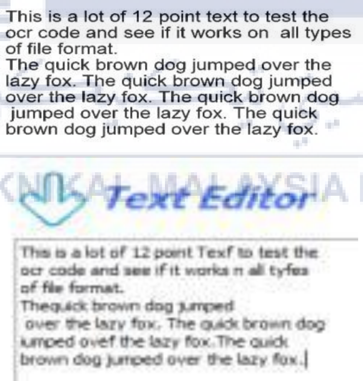
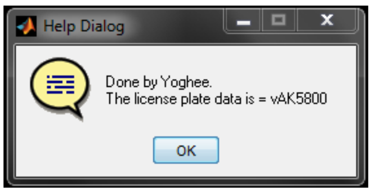
Done by Yoghee.
The license plate data is = PEU2413

(a) Car A (b) Car B (c) Car C

Figure 2.19 Results of Images of Segmented License Plate Undergoing OCR using Bounding Box Feature

2.4 Comparison Previous Research

Table 2.1 Comparison Previous Research

Author's Name	Method	Outcomes	Description
K. Karthick, K. B. Ravindrakumar, R. Francis, and S. Ilankannan	Histogram Equalization		This technique can be used to normalise fluctuations in illumination in image interpretation difficulties.
Y. Sun, X. Mao, S. Hong, W. Xu, and G. Gui	Template-matching		This technique uses template matching to intelligently recognise invoice information and get the relevant data.
N. Sahu and M. Sonkusare	Magnetic Character Recognition (MCR)	 Noisy Input Optical Character Recognition Sentence Boundary Detection Tokenization Part-of-Speech Tagging Error Present in Stage SB T POS Photocopy Fax Scan Document Latent Errors	The MCR approach is commonly used to recognise patterns or alphabets.
S. Desai and A. Singh	Template-matching and Back Propagation	This is a lot of 12 point text to test the ocr code and see if it works on all types of file format. The quick brown dog jumped over the lazy fox. The quick brown dog jumped over the lazy fox. The quick brown dog jumped over the lazy fox. The quick brown dog jumped over the lazy fox. This is a lot of 12 point Texf to test the ocr code and see if it works n all tyfes of file format. Thequick brown dog jumped over the lazy fox. The quick brown dog iumped ovef the lazy fox.The quick brown dog jumped over the lazy fox.	To initiate mistake-free recognition of words from the provided image as input which will assist in document scanning with avoidance to the handwritten text recognition.
K. Yogheedha, A. S. A. Nasir, H. Jaafar, and S. M. Mamduh	Bounding box feature	 Done by Yoghee. The license plate data is = vAK5800 	Analyze the characters printed on the segments licence plate image and generate character-based data produces.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter aims to talk about methods that came to be utilised to track the project's development. This chapter explains how the suggested Optical Character Recognition (OCR) system algorithm is implemented in software, including the work process, methodology, and techniques used. The flow chart was provided to highlight the overall plan for this project's development. The design, data gathering, and process flow chart were all part of the methodology.

3.2 Methodology

UTeM UNIVERSITI TEKNIKAL MALAYSIA MELAKA

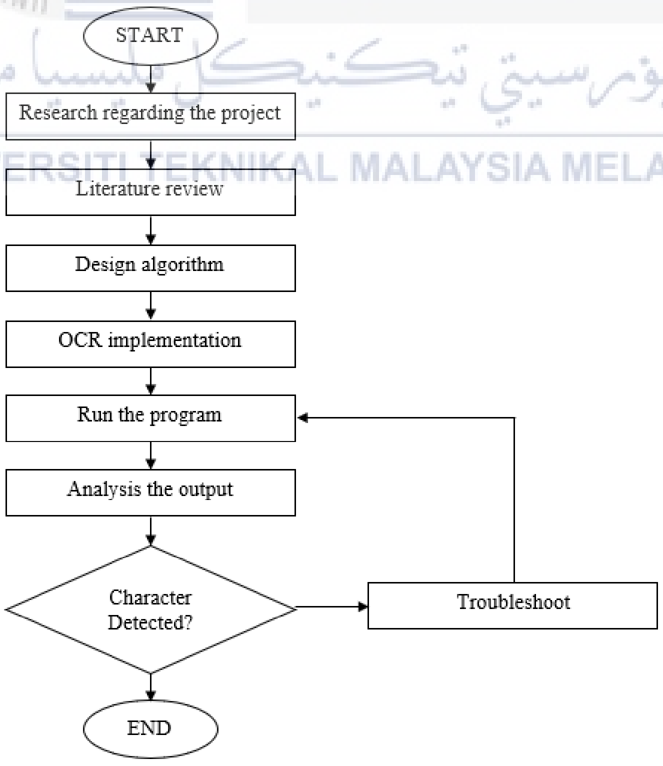


Figure 3.1 Planning Project Flow

The planning project flow is depicted in Figure 3.1 as a flowchart. The first step is to conduct some preliminary research on the project. It also includes a literature review, which is discussed concerning a previous project and a current project. The literature review that was analysed in Chapter 2 provided a brilliant idea for gathering information for the project. Then, a suitable algorithm was built to implement OCR in the employed software. Then, the coding based on the algorithm will be constructed and the programme will be executed thus the output will be displayed. If the output does not match the existing criteria, the application will troubleshoot and execute to fix the problem. Finally, the application will repeat the procedure until it is completed, and it will continue to run without errors.

3.3 Flowchart Represents the Process of the Project

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

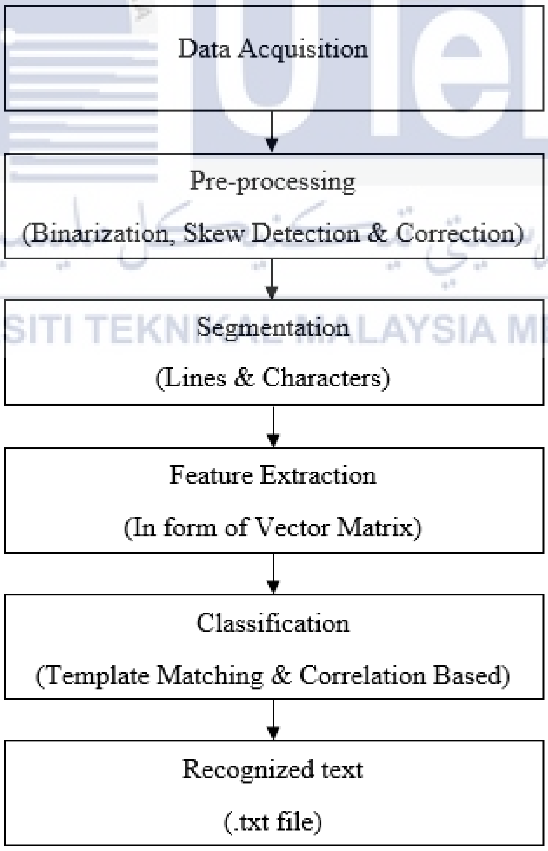


Figure 3.2 Project Development Flowchart

The project's flow is depicted in Figure 3.2 as a flowchart, from the beginning to the end. The initial phase is data collection, after which an image can be made using the font that has been tested. The photograph will be pre-processed to increase its quality before going through the rest of the procedure. The text will then be divided into lines, words, and characters. As soon as the photos have completed the last procedure, the vital information or feature of each segmented character will be extracted depending on the individual character. The character was then categorised by applying the template matching and picture correlation methods to match the existing template [18]. If an error is detected, the application will start troubleshooting. The character will then be extracted as a text file that can be edited in a notepad.

3.4 Block Diagram of Project

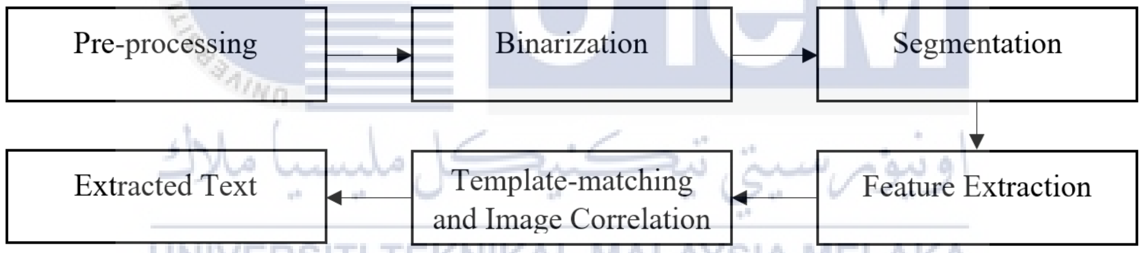


Figure 3.3 Block Diagram of Project

The OCR methodology was developed using the block diagram illustrated in Figure 3.3 as a foundation. The image will go through a pre-processing step in which it will be converted to a binary picture and noise will be removed. The character will be segmented after that. Following that, the individual features of each segmented character will be retrieved as matrixes. After that, utilising template-matching and image correlation methods, the character will be matched to a set prototype for each class for categorization [19]. Finally, all of the collected characters from the test will be analysed for accuracy.

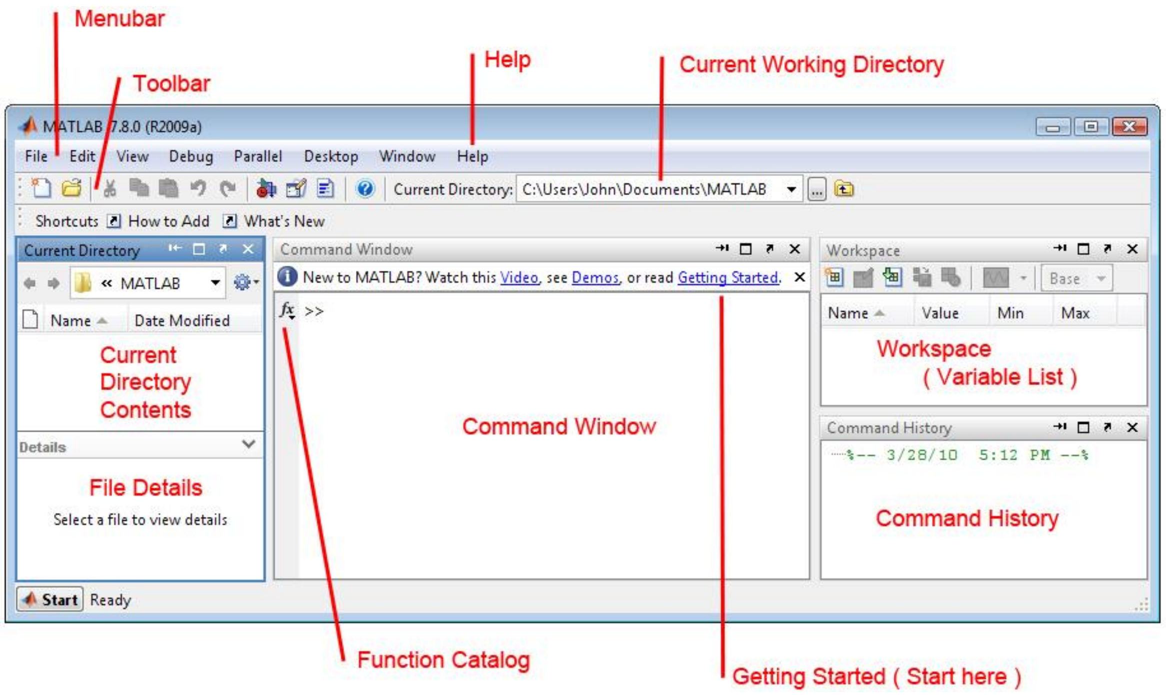
3.5 Software Implementation

The abbreviation MATLAB stands for MATrix LABoratory. People can use MATLAB to solve a wide range of analytical and numerical problems using the matrix method [20]. MATLAB is high-performance computing, visualisation, and programming environment used in engineering, scientific and technical computing, and programming.

MATLAB is both a high-performance technical computer language and a very capable programming language with a large number of toolboxes [21]. It combines computation, visualisation, and programming in a user-friendly environment where the problem and solution are displayed in familiar notation. As a result, one of the main reasons for adopting this programme is the ease with which it can debug and build computational codes. An image processing toolkit was used to process the image for this project. For image manipulation, inspection, visualisation, and method expansion, the Image Processing Toolbox include a thorough scale of industry-standard algorithms and progress mechanisms. Picture segmentation, image enhancement, noise depletion, geometric modification, image registration, and 3D image processing are all possible with this toolbox.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

There are five important windows; the command window, which allows you to type in running commands, the command history window, which shows the passkey in data, workspace, which displays a list of currently defined variables, the current directory window, which unveils the contents of the present working directory, and the file features window, which shows full descriptions of the folders in the contemporary working directory [22]. Figure 3.4 depicts the most significant aspects of the MATLAB programming environment, whereas Figure 3.5 depicts the operator within MATLAB.


Menubar
Toolbar
Help
Current Working Directory
MATLAB 7.8.0 (R2009a)
File Edit View Debug Parallel Desktop Window Help
Current Directory: C:\Users\John\Documents\MATLAB
Shortcuts How to Add What's New
Current Directory
MATLAB
Name Date Modified
Current Directory Contents
Details
File Details
Command Window
Workspace
Base
Name Value Min Max
Workspace (Variable List)
Command History
Command Window
Command History
Start Ready
Function Catalog
Getting Started (Start here)

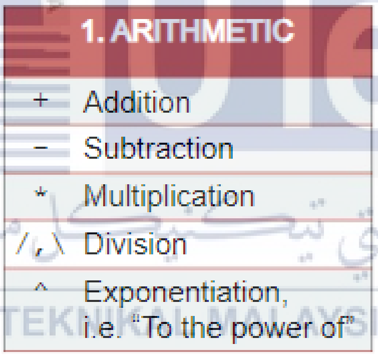
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>>
```

```
%-- 3/28/10  5:12 PM --%
```

Select a file to view details

Figure 3.4 MATLAB Work Environment



1. ARITHMETIC	
+	Addition
−	Subtraction
*	Multiplication
/, \	Division
^	Exponentiation, i.e. "To the power of"

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.5 MATLAB Operation

3.6 Optical Character Recognition

Optical Character Recognition (OCR) systems are a quick and efficient approach to sharing the material within images. Otherwise, it will take a long time to manually enter the text into the images [23]. Several strategies are applied to the systems to obtain precise output for character recognition to utilise these strategies.

The technology allows the OCR system to extract text from images to access the information contained within them [24]. Recent papers have focused on improving methodologies and adding new methods to the OCR to increase performance in the future. OCR techniques are useful and are becoming more popular in the realm of communication and multimedia.

3.6.1 Pre-processing

The digital image must be pre-processed because it may lack picture resolution, making it difficult to see the text. Pre-processing is a technique for improving image quality when the scanned image is of poor quality [25]. Several processes can be applied to the image in this regard. The image will go through a noise removal and binarization process, in which the noise will be removed. Binarization is a technique that uses thresholding to convert a digital image into a grayscale image. As illustrated in Figure 3.6, the image must be converted from RGB (red, green, and blue) colour to a binary image. Because pixels are stored in bits, the present image will only have two colours: black (0) and white (1). Thresholding is an image segmentation technique that distinguishes objects by transforming grayscale photos into binary pictures. As a result, writing stands out from the background considerably more easily. The image quality will be increased as a result of this process.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

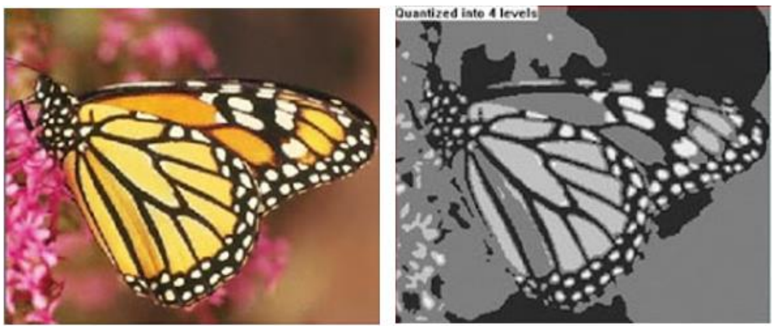


Figure 3.6 Image after Binarization

3.6.2 Segmentation

To move on to the next step, the segmentation approach divided the image into subparts. External segmentation and internal segmentation are the two forms of sectionalisation [26]. Internal segmentation will isolate the characters and letters, while external segmentation will isolate the phrases and paragraphs. Line segmentation, word segmentation, and character segmentation are the three varieties of segmentation.

3.6.2.1 Line Segmentation

Only the lines are extracted and differentiated inside the images during line segmentation. As a result, the lines can be retrieved by projecting an image horizontally [27]. Horizontal projection can be used to divide lines that are well separated and are not tiled that operate as text separators, as shown in Figure 3.7. The location of the lines' boundaries is easily specified and can be utilised to evaluate these valleys. The following equation is used to obtain the horizontal projection of binary objects:

$$h_i(x) = \sum_{x=0}^{M-1} O_i(x, y)$$

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.2.2 Word Segmentation

The process of segmenting a string into its constituent words is known as word segmentation. Word splitting is a way of processing concatenated text to determine where the word breaks occur [28]. Lines can be divided by seeking for minima in the page's horizontal projection profile, while terms can be detached by seeking for minima in the perpendicular projection profile of a sole row. Words can be split from the line and individual letters extracted from the word utilising the hollows in the vertical projection of a line picture, as shown in Figure 3.7. The following equation is used to obtain the vertical projection of binary objects:

$$v_i(y) = \sum_{y=0}^{N-1} O_i(x, y)$$

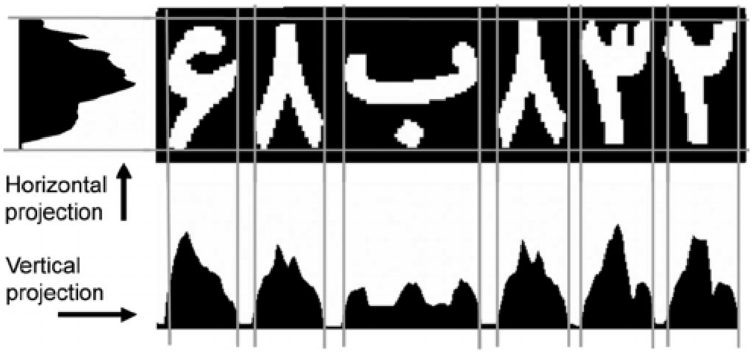


Figure 3.7 Horizontal and Vertical Projection

3.6.2.3 Character Segmentation

Character segmentation is the mass effective tread for OCR systems because the characters are separated into the smallest unit [29]. Characters from a single word can be retrieved using vertical projection in character segmentation. Character segmentation, as shown in Figure 3.8, is a tough phase in OCR systems since it extracts significant portions for study. Poor segmentation method leads to incorrect recognition or rejection after pre-processing.

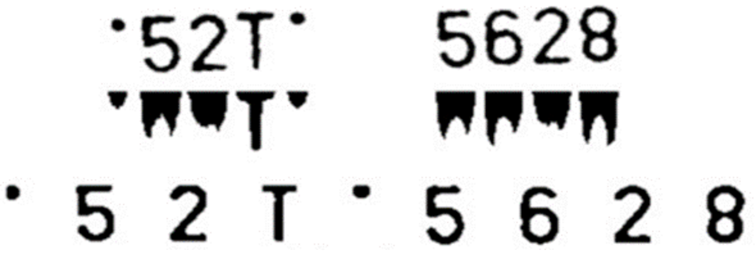
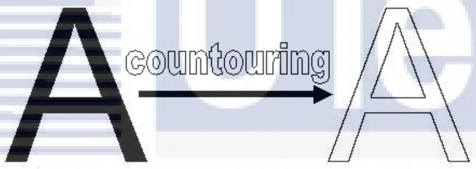


Figure 3.8 Correct Character Segmentation

UTeM UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.3 Feature Extraction

The character may be incorrectly categorised after going through the segmentation procedure. This occurs because the character shapes are nearly identical, and the selected features are insufficient to distinguish the various classes, or because the features are difficult to determine and were calculated incorrectly. As shown in Figure 3.9, feature extraction decomposes a character into features such as lines, line direction, and line intersections. Feature Extraction has two goals: to extract attributes that can uniquely identify a character and to distinguish between similar characters [30]. As a result, individual lines and characters are isolated and identified, and then picture correlation is used to compare them to the template [31]. Image correlation function based on features collected from characters used in classification for feature extraction.



اونيۆرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.9 Feature Extraction of Character A

3.6.4 Feature Template-matching and Image Correlation

The previous output data must be classified so that the right classification of each piece of data may be determined. This function was used to match the extracted character to the set of character prototypes that each class represents [32]. In the meantime, the gap in the middle of the design and each prototype is determined, and the prototype class that provides the finest fixture in allocating to the pattern is determined. The extracted character from the previous phase will match the existing prototype, as seen in Figure 3.10.

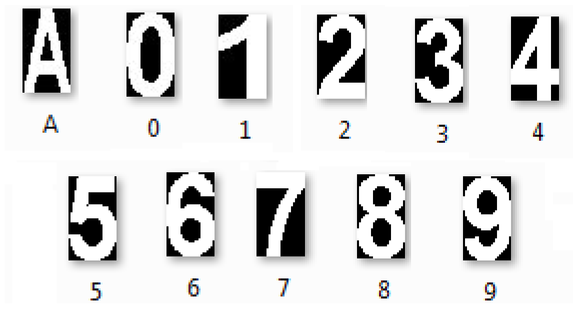


Figure 3.10 Templates for Text Recognition

Image correlation was utilised to measure the degree to which two variables matched to match the templates. The equivalent pixel merits in two images, form and origin, are the two variables. Corr2 is a MATLAB function that calculates the correlation coefficient. This can be demonstrated using the following formula [33]:

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}$$

Where,

A = the template grey level image,

$\bar{A}$ = the average grey level in the template image,

B = the source image section,

$\bar{B}$ = the average grey level in the source image.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.7 Summary

This chapter presents the proposed methodology to develop a new, effective and integrated approach in increasing higher accuracy while detecting character by using template-matching and image correlation [34]. All methods are needed use to achieve the target by using OCR.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

The outcomes of the data analysis based on the case study samples are presented in this chapter. The different font images are the primary source of data. The findings will be presented concerning the study's goals. The method that was utilised to analyse the results was previously detailed in the methodology chapter.

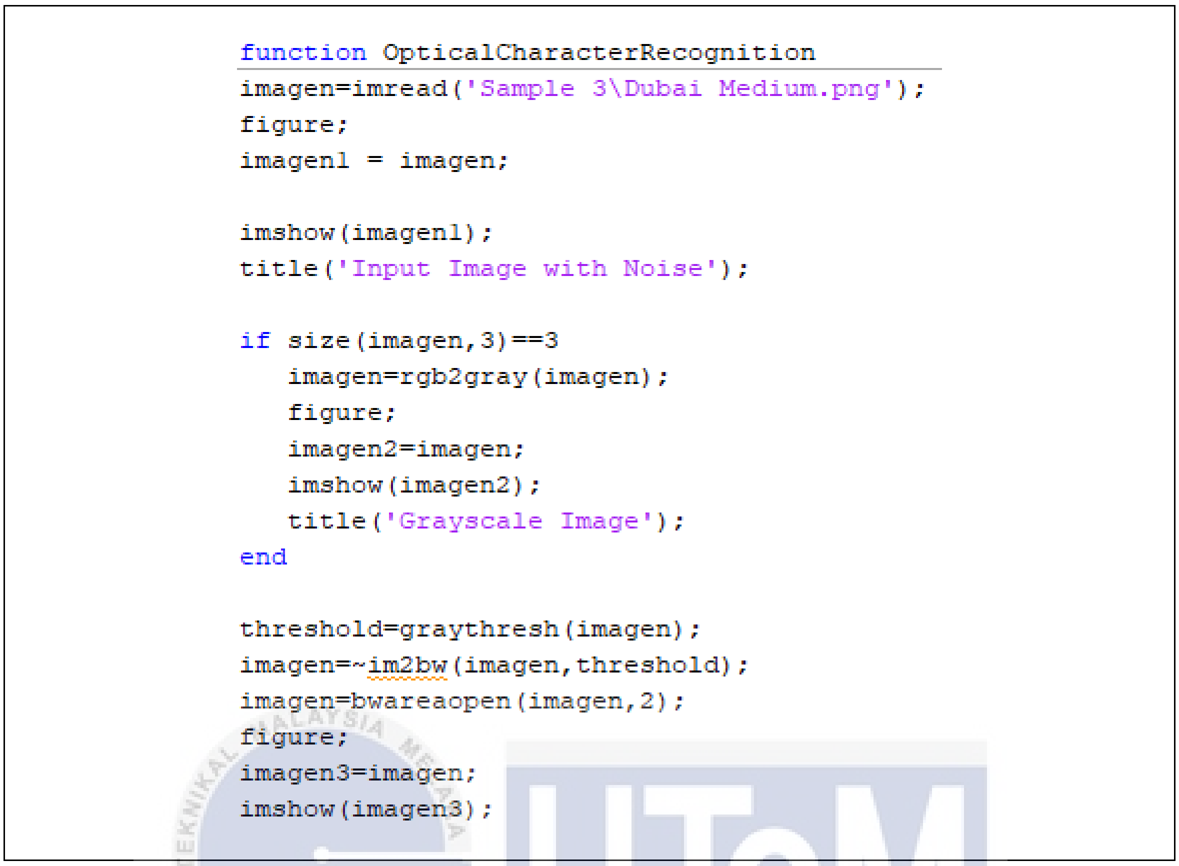
4.2 Software Simulation

As previously stated, this research relies solely on computer simulations conducted in MATLAB, with no external hardware devices used. In the end, the simulation of this project will be shown and explained in further detail.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2.1 Coding of the System

This is the code that will process the image first before extracting the text. The user can then add more coding and modify the code to match the design requirements. Regardless, this project necessitates the use of some code.



```
function OpticalCharacterRecognition
imagen=imread('Sample 3\Dubai Medium.png');
figure;
imagen1 = imagen;

imshow(imagen1);
title('Input Image with Noise');

if size(imagen,3)==3
    imagen=rgb2gray(imagen);
    figure;
    imagen2=imagen;
    imshow(imagen2);
    title('Grayscale Image');
end

threshold=graythresh(imagen);
imagen=~im2bw(imagen,threshold);
imagen=bwareaopen(imagen,2);
figure;
imagen3=imagen;
imshow(imagen3);
```

Figure 4.1 The code for image pre-processing in MATLAB

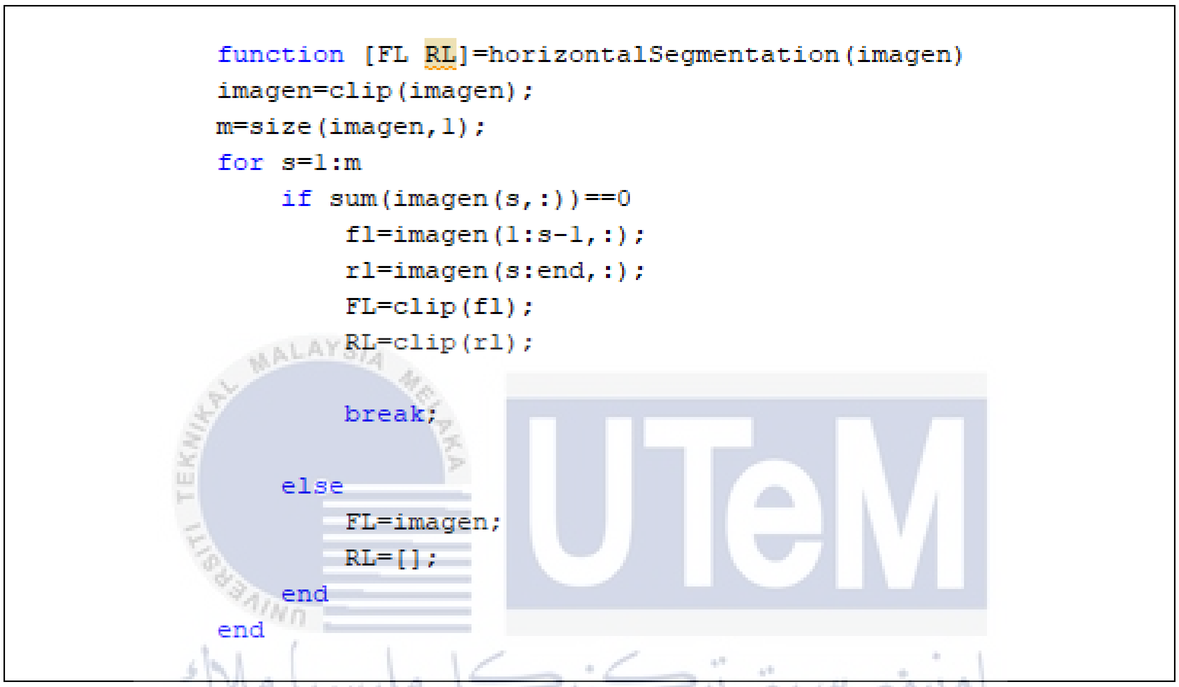
Figure 4.1 shows how an image from sample 3 with the font type Dubai Medium was imported into MATLAB. The original image with noise will be shown in the first figure. Because sample 3 is a red-coloured font image, it will be reduced to grayscale before being translated to binary. The image can be used to convert an intensity image to a binary image within the range [0 1] by utilising graythresh.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
threshold          0.5314
```

The value indicated that the value is optimal, which is normalised between the foreground and background values of [0 1]. After binarization, the image will be processed for noise removal using the code in line 17, which will remove all objects with less than 20 pixels, which is the minimum pixel size. The image will be segmented after it has been pre-processed.

Horizontal segmentation was used to check for a line break in the image, as shown in Figure 4.2. Words are detected line by line, beginning with the first line and ending with the last line [35]. The text will be clipped horizontally in the form of a matrix using the clip function. As a result, the initial line was clipped out of the horizontal segmentation till the last line.

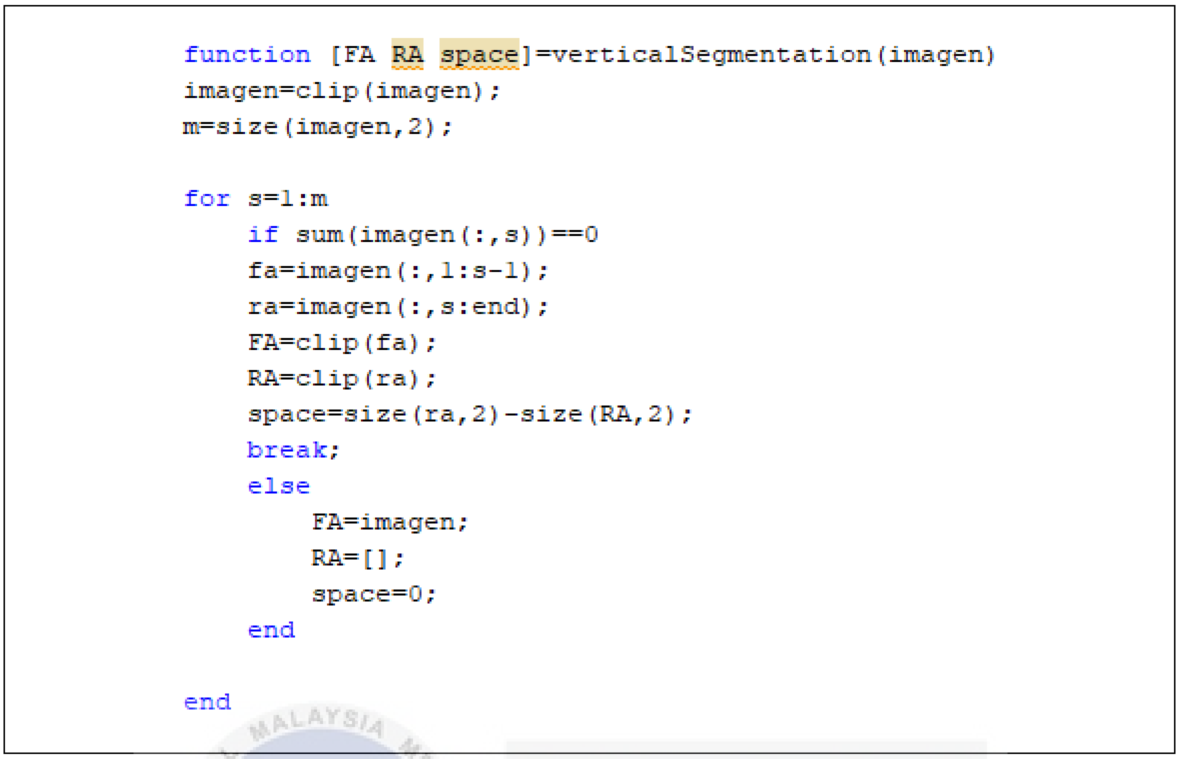


```
function [FL RL]=horizontalSegmentation(imagen)
imagen=clip(imagen);
m=size(imagen,1);
for s=1:m
    if sum(imagen(s,:))==0
        fl=imagen(1:s-1,:);
        rl=imagen(s:end,:);
        FL=clip(fl);
        RL=clip(rl);

        break;

    else
        FL=imagen;
        RL=[];
    end
end
```

Figure 4.2 The code for image horizontal segmentation in MATLAB

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Horizontal segmentation was used to check for a line break in the image [36], as shown in Figure 4.2. Words are detected line by line, beginning with the first line and ending with the last line. The text will be clipped horizontally in the form of a matrix using the clip function. As a result, the initial line was clipped out of the horizontal segmentation till the last line.

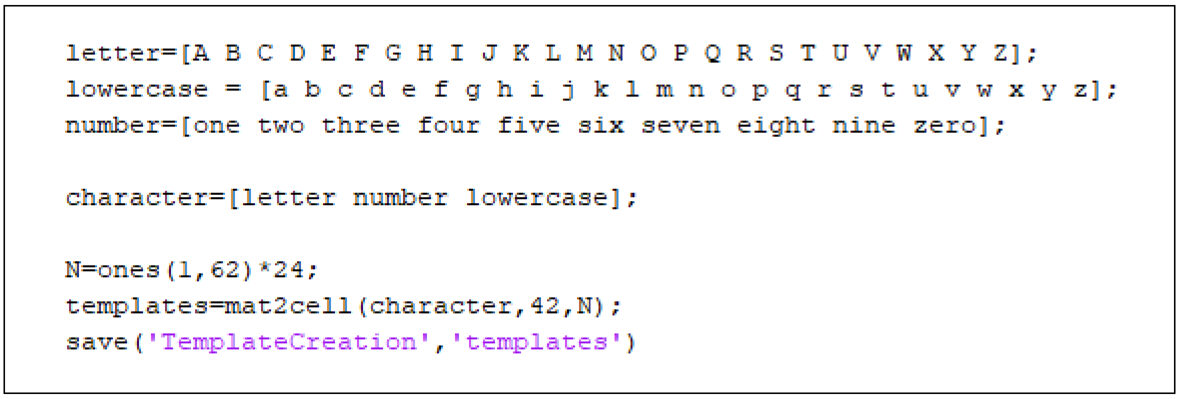


```
function [FA RA space]=verticalSegmentation(imagen)
imagen=clip(imagen);
m=size(imagen,2);

for s=1:m
    if sum(imagen(:,s))==0
    fa=imagen(:,1:s-1);
    ra=imagen(:,s:end);
    FA=clip(fa);
    RA=clip(ra);
    space=size(ra,2)-size(RA,2);
    break;
    else
        FA=imagen;
        RA=[];
        space=0;
    end

end
```

Figure 4.3 The code for image vertical Segmentation in MATLAB

Vertical segmentation checked the line break between the characters [37], as seen in Figure 4.3. Characters are recognised one by one, beginning with the first letter and ending with the last alphabet, while also discovering space between the first and last alphabets. The clip function will be vertically clipped out in the form of a matrix. As a result, the first character is used until the last character is cut out of the vertical segmentation.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

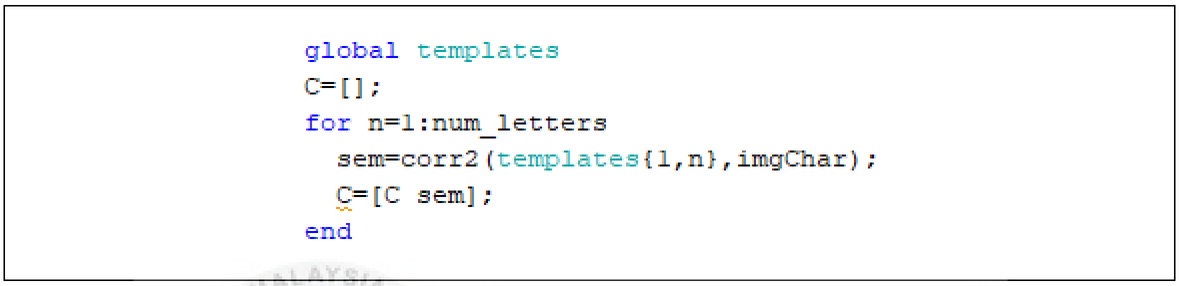


```
letter=[A B C D E F G H I J K L M N O P Q R S T U V W X Y Z];
lowercase = [a b c d e f g h i j k l m n o p q r s t u v w x y z];
number=[one two three four five six seven eight nine zero];

character=[letter number lowercase];

N=ones(1,62)*24;
templates=mat2cell(character,42,N);
save('TemplateCreation','templates')
```

Figure 4.4 The code for template creation in MATLAB

The template was introduced as described in appendix 5 and 6 based on Figure 4.4. Then, as indicated in line 4, all the matrices are placed into a new variable called characters.

The template was inserted as described in the appendix based on Figure 4.4. All capital, lowercase, and numeric characters are converted to the letter, lowercase, and numeric characters in the form of a matrix. Then, as indicated in line 4, all the matrices are placed into a new variable called characters. The character matrix is recorded in 42*24 dimensions in each cell, which refers to the dimension of a normalised character picture. The total number of characters is 62, with 26 in lowercase, 26 in uppercase, and 10 in numerals.



```
global templates
C=[];
for n=1:num_letters
  sem=corr2(templates{1,n},imgChar);
  C=[C sem];
end
```

Figure 4.5 The code for template matching in MATLAB

Image correlation was utilised to match the template with the font image that had already been segmented in segmentation [38], as shown in Figure 4.5. For further information, see appendix 7 and appendix 1. Refer to Appendix 7, where the segmented character will correspond to the uppercase, numeral, and then lowercase. As a result, the text will be extracted after the character has identified a match within the template. Then, depending on the correctness of the extracted font with the image, an analysis was performed to determine which fonts corresponded to the existing template.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3 Simulation Result

4.3.1 Image Processing

Cambria, Lucida Sans, Dubai Medium, Microsoft YaHei, Malgun Gothic, and Lato are among the fonts used in this image. The results of image preprocessing are shown in the figures below.

Table 4.1 Preprocessing Image

Sample	Font	Input Image	Preprocessing Image
Sample 1	Cambria	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL
	Lucida Sans	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL
	Dubai Medium	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL
	Microsoft YaHei	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL
	Malgun Gothic	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL
	Lato	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL
Sample 2	Cambria	old ways do not open new doors	old ways do not open new doors
	Lucida Sans	old ways do not open new doors	old ways do not open new doors
	Dubai Medium	old ways do not open new doors	old ways do not open new doors

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	Microsoft YaHei	old ways do not open new doors	old ways do not open new doors
	Malgun Gothic	old ways do not open new doors	old ways do not open new doors
	Lato	old ways do not open new doors	old ways do not open new doors
Sample 3	Cambria	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE IS TOO SHORT TO LIMIT YOURSELF
	Lucida Sans	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE IS TOO SHORT TO LIMIT YOURSELF
	Dubai Medium	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE IS TOO SHORT TO LIMIT YOURSELF
	Microsoft YaHei	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE IS TOO SHORT TO LIMIT YOURSELF
	Malgun Gothic	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE IS TOO SHORT TO LIMIT YOURSELF
	Lato	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE IS TOO SHORT TO LIMIT YOURSELF
	Cambria	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Sample 4	Lucida Sans	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
	Dubai Medium	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
	Microsoft YaHei	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
	Malgun Gothic	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9
	Lato	0 1 2 3 4 5 6 7 8 9	0 1 2 3 4 5 6 7 8 9

4.3.2 Sample 1

This is an image with all uppercase letters and 39 alphabets in a single sentence.

Table 4.2 Text Extraction of Sample 1

Font	Sample	Extracted Text
Cambria	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KILLs MORE DREAMs THAN FAILURE EVERwILL
Lucida Sans	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KlLLs MORE DREAMs THAN FAlLURE EVERwlLL
Dubai Medium	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KlLLs MORE DREAMs THAN FAlLURE EVERWlLL
Microsoft YaHei	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KlLLs MORE DREAMs THAN FAlLURE EVERWlLL

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Malgun Gothic	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KlLLs MORE DREAMs THAN FAlLURE EVERWlLL
Lato	DOUBT KILLS MORE DREAMS THAN FAILURE EVER WILL	DOUBT KlLLs MORE DREAMs THAN FAlLURE EVERWlLL

4.3.3 Sample 2

These are the images with all lowercase font and 24 alphabets in a single sentence.

Table 4.3 Text Extraction of Sample 2

Font	Sample	Extracted Text
Cambria	old ways do not open new doors	OId waYs dO nOt Open new dOOrs
Lucida Sans	old ways do not open new doors	Old ways dO nOt Open newdOOrs
Dubai Medium	old ways do not open new doors	odd Ways do not open newdoors
Microsoft YaHei	old ways do not open new doors	Oid Ways dO nOt Open neW dOOrs

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Malgun Gothic	old ways do not open new doors	Old Ways dO nOt Open meW dOOrs
Lato	old ways do not open new doors	Old Ways dO nOt Open neW dOOrs

4.3.4 Sample 3

These are the photographs where the font was completely red and there were 29 alphabets in a single sentence.

Table 4.4 Text Extraction of Sample 3

Font	Sample	Extracted Text
Cambria	LIFE IS TOO SHORT TO LIMIT YOURSELF	LfFE fs TOO sHORT TO LfMITYOURsELF
Lucida Sans	LIFE IS TOO SHORT TO LIMIT YOURSELF	LlFE ls TOO sHORT TO LlMlTYOURsELF
Dubai Medium	LIFE IS TOO SHORT TO LIMIT YOURSELF	LlFE lS TOO sHORT TO LlMlTYOURsELF
Microsoft YaHei	LIFE IS TOO SHORT TO LIMIT YOURSELF	LlFE ls TOO sHORT TO LlMlTYOURsELF
Malgun Gothic	LIFE IS TOO SHORT TO LIMIT YOURSELF	LlFE ls TOO sHORT TO LlMlT YOURsELF

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Lato	LIFE IS TOO SHORT TO LIMIT YOURSELF	LIFE ls TOO sHORT TO LlMlTYOURsELF

4.3.5 Sample 4

These are the images with all of the typeface in numbers and a single word with 10 characters.

Table 4.5 Text Extraction of Sample 4

Font	Sample	Extracted Text
Cambria	0 1 2 3 4 5 6 7 8 9	O T 2 3 4 s 6 7 8 9
Lucida Sans	0 1 2 3 4 5 6 7 8 9	O 1 2 34s 6789
Dubai Medium	0 1 2 3 4 5 6 7 8 9	o I 2 3 4 5 6 7 8 9
Microsoft YaHei	0 1 2 3 4 5 6 7 8 9	O 1 ZB4s 6789
Malgun Gothic	0 1 2 3 4 5 6 7 8 9	O 1 2 B4 s 6789
Lato	0 1 2 3 4 5 6 7 8 9	O I 2 3 4 5 5 7 8 9

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.4 Result Analysis

4.4.1 An Analysis Based on Accuracy of Text Extraction

An analysis of the text accuracy based on the text extraction based on four separate samples. The accuracy is measured in per cent (%) and includes both right and incorrect recognition. This precision was calculated using the following formula:

$$Accuracy = \frac{Correctly\ detected\ words}{Total\ words} \times 100$$

Table 4.6 Percentage of Accuracy for Correct Recognition in Text Extraction

Test Sample	Font	Correct Recognition	Incorrect Recognition	Accuracy (%)
Sample 1	Cambria	36	3	92
	Lucida Sans	33	6	85
	Dubai Medium	34	5	87
	Microsoft YaHei	34	5	87
	Malgun Gothic	34	5	87
	Lato	34	5	87
Sample 2	Cambria	16	8	67
	Lucida Sans	18	6	75
	Dubai Medium	22	2	92
	Microsoft YaHei	15	9	63
	Malgun Gothic	15	9	63
	Lato	16	8	67
Sample 3	Cambria	23	6	79
	Lucida Sans	22	7	76
	Dubai Medium	23	6	79
	Microsoft YaHei	22	7	76
	Malgun Gothic	22	7	76
	Lato	22	7	76

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Sample 4	Cambria	7	3	70
	Lucida Sans	8	2	80
	Dubai Medium	8	2	80
	Microsoft YaHei	6	4	60
	Malgun Gothic	7	3	70
	Lato	7	3	70

4.4.1.1 Sample 1

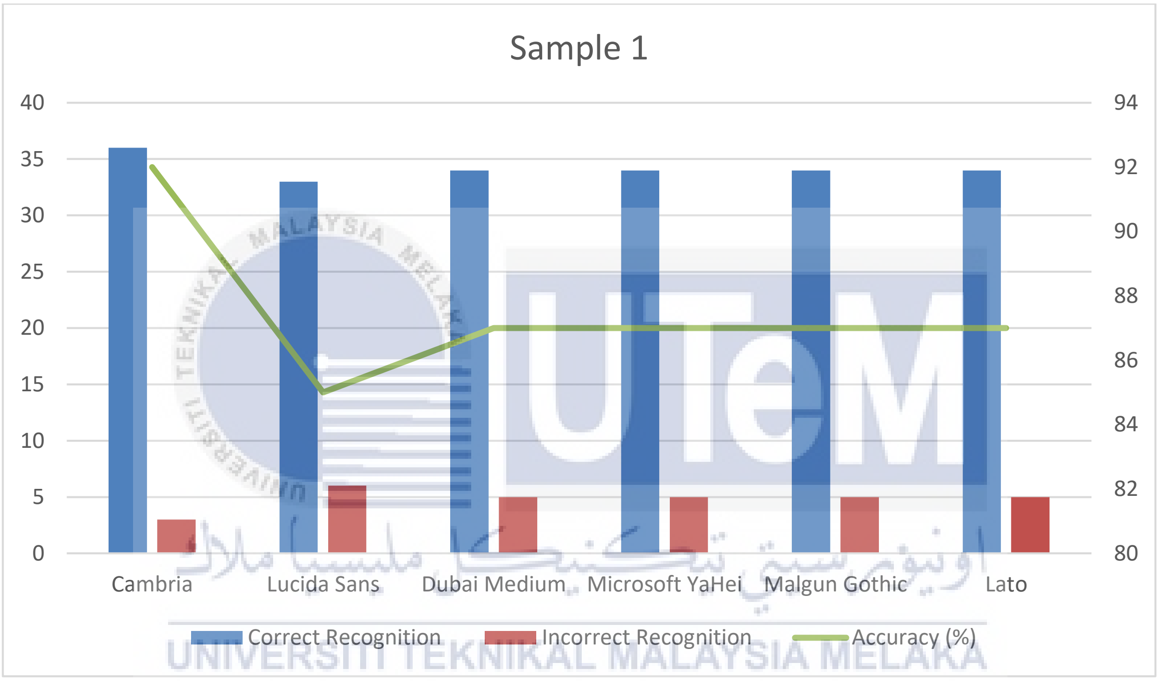


Figure 4.6 Bar Chart of Result Analysis for Sample 1

Cambria (36 alphabets), Lucida sans (33 alphabets), Dubai Medium, Microsoft YaHei, Malgun Gothic and Lato (34 alphabets) are the fonts correctly recognised in Table 7 (39 alphabets). Using the combination chart presented in Figure 4.6, the highest font accuracy is Cambria (92%), which has the lowest wrong recognition rate (3 alphabets). Following that are Dubai Medium, Microsoft YaHei, Malgun Gothic and Lato (87%), which have the same inaccurate recognition (5 alphabets), while Lucida Sans (85%) has the highest incorrect recognition (6 alphabets).

4.4.1.2 Sample 2

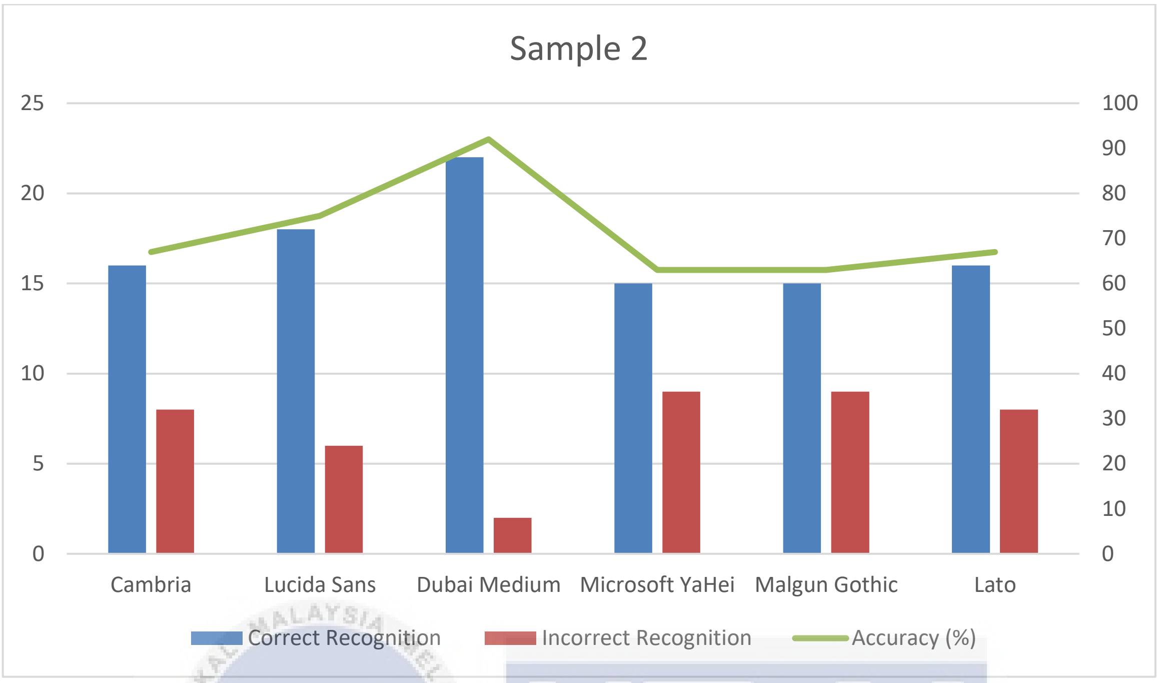


Figure 4.7 Bar Chart of Result Analysis for Sample 2

Cambria and Lato (16 alphabets), Microsoft YaHei and Malgun Gothic (15 alphabets), Dubai Medium (22 alphabets), and Lucida Sans (18 alphabets) are the fonts correctly recognised in Table 4.6. Using the combination chart given in Figure 4.7, the font with the highest accuracy (92%) and the fewest wrong recognitions (2 alphabets) is Dubai Medium, followed by Lucida Sans (75%) with the second-highest correct recognition (18 alphabets). Cambria and Lato (67%) have the same incorrect recognition (8 alphabets) and finally Microsoft YaHei and Malgun Gothic (63%) with the highest improper recognition (9 alphabets).

4.4.1.3 Sample 3

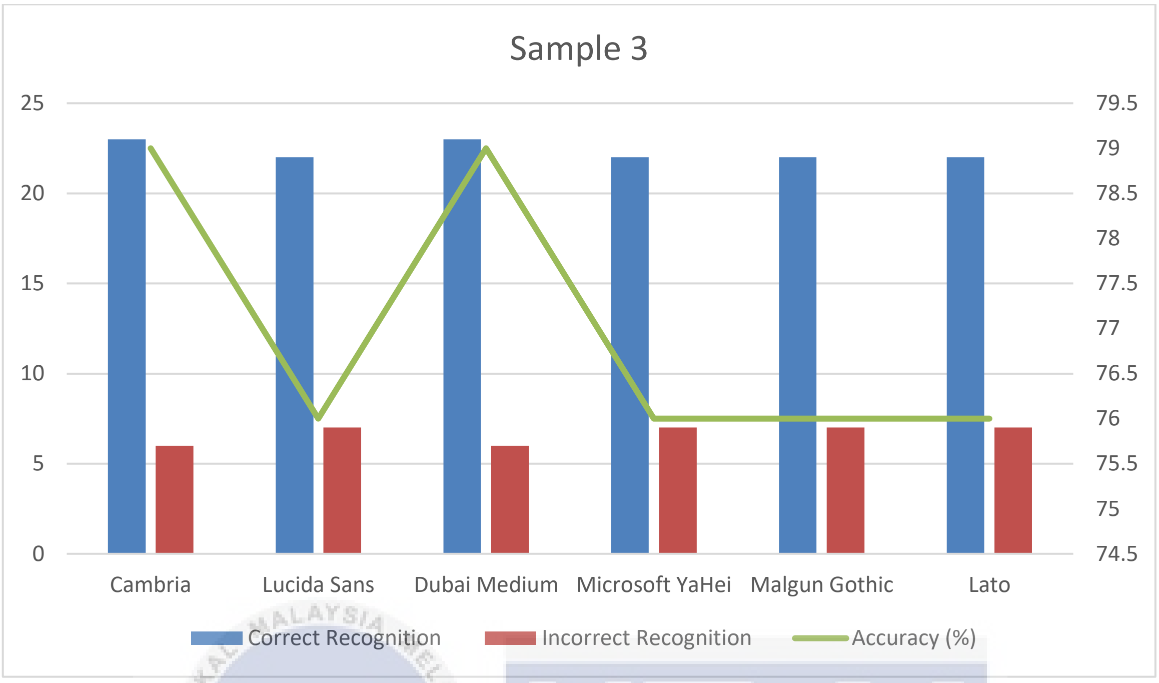


Figure 4.8 Bar Chart of Result Analysis for Sample 3

From the table, the fonts Cambria and Dubai Medium (23 alphabets), Lucida Sans, Microsoft Yahei, Malgun Gothic and Lato (22 alphabets) are correctly recognised in the test. Using the combination chart presented in Figure 4.8, the font with the highest accuracy (79%) and correct recognition (23 alphabets) with the fewest wrong recognitions (6 alphabets) are Cambria and Dubai Medium. The remaining result has inaccurate recognition (7 alphabets) which are Lucida Sans, Microsoft Yahei, Malgun Gothic and Lato (76%) with the second most correct recognition (22 alphabets).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.4.1.4 Sample 4

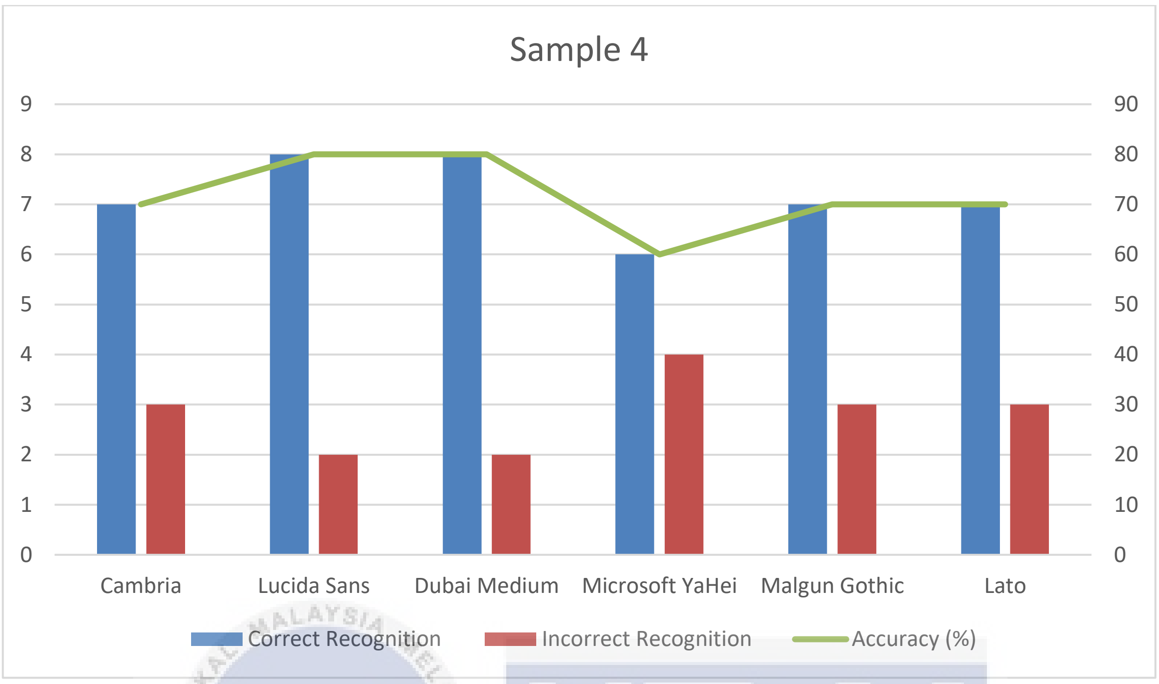


Figure 4.9 Bar Chart of Result Analysis for Sample 4

From Table 4.6, the correct recognition of the fonts is Lucida Sans and Dubai Medium (8 numbers), Cambria, Malgun Gothic and Lato (7 numbers) and Microsoft YaHei (6 numbers). By using a combination chart as shown in Figure 4.9, the highest accuracy of font is Lucida Sans and Dubai Medium (80%) that have 2 numbers of incorrect recognition. Then, followed by Cambria, Malgun Gothic and Lato (70%) that have the same incorrect recognition (3 numbers) and lastly, Microsoft YaHei (60%) has the highest incorrect recognition (4 numbers).

4.4.2 Analysis Based on Sum of Accuracy

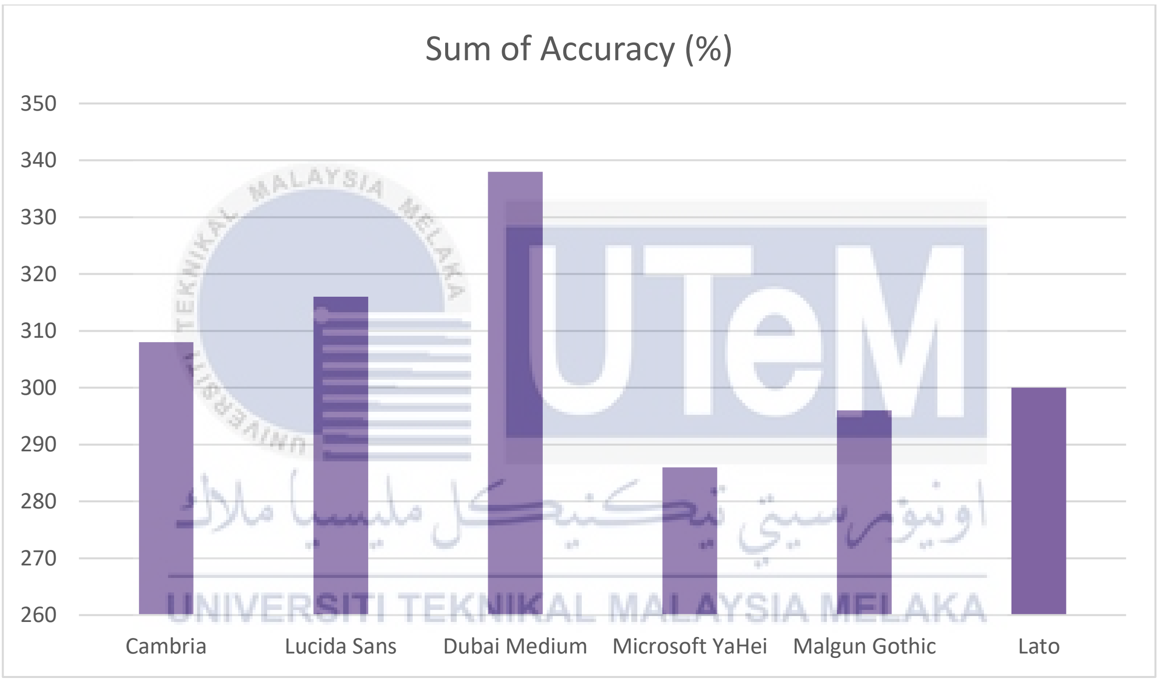
Each typeface displayed a varying level of accuracy concerning the template. The table below shows the sum of accuracy for all samples to determine which typeface had the highest accuracy. Using the formula below, the calculation of the total accuracy:

$$Sum\ of\ Accuracy\ (\%) = Accuracy\ (Sample\ 1 + Sample\ 2 + Sample\ 3 + Sample\ 4)$$

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 4.7 Sum of Accuracy

Font	Sum of Accuracy (%)
Cambria	308
Lucida Sans	316
Dubai Medium	338
Microsoft YaHei	286
Malgun Gothic	296
Lato	300



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.10 Bar Chart of Sum of Accuracy

The font that had the highest accuracy is Dubai Medium (338%) followed by Lucida Sans (316%), Cambria (308%), Lato (300%), Malgun Gothic (296%) and lastly Microsoft YaHei (286%). Out of the 6 selected fonts, Dubai Medium had the highest success in detecting fonts in 4 different types of samples. It can be concluded that the font was the most compatible with the existed template.

Above all, the experimental result was a Sans Serif font with no curls and a proportionally spaced font that was compatible with the existing template. Using the OCR technique, a large number of fonts in Microsoft Word were searched and tested. The font was created in Microsoft Word, and the fonts were screenshotted and saved as an image. Six fonts were chosen from all of the fonts based on criteria that precisely matched the template. Font size, colour, contrast, and density were among the factors that could influence accuracy. The font size used in the sample is 24. In terms of colour, the fonts evaluated in sample 4 have a red background. Dubai Medium had the highest contrast and density of the six typefaces.

This OCR technology had several flaws. The most difficult part of this OCR system was distinguishing between characters that looked similar. The system, for example, sought to match O with 0 (number 0), l (lowercase L) with 1 (number 1), S (uppercase s) with 5 (number 5) and l (lowercase L) with I. (uppercase i). The accuracy was also affected by the number of words used. The less accurate the typefaces are in detecting the font for extraction, the longer the words are. Some of the 26 characters were misidentified as a result of the words' chosen characters. Image processing was also used to detect text. The lower the accuracy in identifying the text, the bigger the image noise.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.5 Summary

This chapter contains findings that address the study's three research questions, followed by a discussion of the findings. The image will first go through image preprocessing, which will transform it into a binary image and remove the noise. The image will then be split and the typeface will be matched to the template. The fonts will then be analysed depending on the correct identification character, which will be output onto the notepad. In comparison to the other fonts, Dubai Medium had the highest accuracy, according to the results.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

. To complete this report, this chapter ended all results, analyses, observations, and accuracy evaluations. Due to this problem, future work proposals were offered.

5.2 Conclusion

Finally, it can be stated that using template-matching and picture correlation, text extraction based on distinct fonts within photos may be implemented [39]. As a result, the project's objectives were met in full. The text can be successfully extracted based on a specific typeface with the maximum frequency utilising this method. Despite this, the text can be edited because it will be displayed in a notepad. Furthermore, the user does not need to be concerned if there is a misdetection alphabet because it can be manually corrected.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

As a result, the text extraction accuracy was also checked by computing the % of right recognition and monitoring the data analysis. In comparison to other fonts, the accuracy of Dubai Medium was found to be the highest. By comparing the input text image with the text extraction output, the right recognition was identified. All of the typefaces displayed varying degrees of accuracy concerning the existing template. After successfully finishing this project by referring and picking data from prior related research based on the approach method, the overall result is that all of the fonts correspond to the template.

5.3 Future Works

Based on this study, some improvements and development can be made to the OCR system. First, various enhancing methods such as histogram equalisation, median filtering, and unsharp mask filtering can be used to improve image quality [40]. As a result, image noise can be minimised and the image can become more clear. The template can be multiplied for the font with the least accuracy which is Microsoft YaHei to match the font accurately. As a result, another test can be performed by selecting a new typeface. Second, another approach for text extraction can be used, such as edge detection. This method isolates or extracts the character features where the boundaries are identified, allowing software to evaluate the image and identify the object. As a result, the accuracy of this procedure will be significantly higher.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

REFERENCES

[1] A. Chaudhuri, K. Mandaviya, P. Badelia, and S. K. Ghosh, *Optical character recognition systems*, vol. 352. 2017.

[2] F. Liu, C. Gong, X. Huang, T. Zhou, J. Yang, and D. Tao, "Robust Visual Tracking Revisited: From Correlation Filter to Template Matching," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2777–2790, 2018, doi: 10.1109/TIP.2018.2813161.

[3] X. Liu, Z. Deng, and Y. Yang, "Recent progress in semantic image segmentation," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 1089–1106, 2019, doi: 10.1007/s10462-018-9641-3.

[4] S. Oron, T. Dekel, T. Xue, W. T. Freeman, and S. Avidan, "Best-Buddies Similarity - Robust Template Matching Using Mutual Nearest Neighbors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1799–1813, 2018, doi: 10.1109/TPAMI.2017.2737424.

[5] K. Karthick, K. B. Ravindrakumar, R. Francis, and S. Ilankannan, "Steps involved in text recognition and recent research in OCR; A study," *Int. J. Recent Technol. Eng.*, vol. 8, no. 1, pp. 3095–3100, 2019.

[6] M. Civera, L. Zanotti Fragonara, and C. Surace, "An experimental study of the feasibility of phase-based video magnification for damage detection and localisation in operational deflection shapes," *Strain*, vol. 56, no. 1, pp. 1–19, 2020, doi: 10.1111/str.12336.

[7] R. E. T. Bae, E. R. Arboleda, A. Andilab, and R. M. Dellosa, "Implementation of template matching, fuzzy logic and K nearest neighbor classifier on philippine banknote recognition system," *Int. J. Sci. Technol. Res.*, vol. 8, no. 8, pp. 1451–1453, 2019.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[8] I. Talmi, "Talmi_Template_Matching_With_CVPR_2017_paper.pdf," 2017.

[9] Y. Sun, X. Mao, S. Hong, W. Xu, and G. Gui, "Template matching-based method for intelligent invoice information identification," *IEEE Access*, vol. 7, pp. 28392–28401, 2019, doi: 10.1109/ACCESS.2019.2901943.

[10] N. Sahu and M. Sonkusare, "A Study on Optical Character Recognition Techniques," *Int. J. Comput. Sci. Inf. Technol. Control Eng.*, vol. 4, no. 1, pp. 01–15, 2017, doi: 10.5121/ijcsitce.2017.4101.

[11] S. Desai and A. Singh, "Optical character recognition using template matching and back propagation algorithm," *Proc. Int. Conf. Inven. Comput. Technol. ICICT 2016*, vol. 2016, 2016, doi: 10.1109/INVENTIVE.2016.7830161.

[12] Y. A. Gerhana, M. F. Padilah, and A. R. Atmadja, "Comparison of Template Matching Algorithm and Feature Extraction Algorithm in Sundanese Script Transliteration Application using Optical Character Recognition," vol. 5, no. 1, pp. 73–80, 2020, doi: 10.15575/join.v5i1.580.

[13] K. Karthick, K. B. Ravindrakumar, R. Manikandan, and R. Cristin, "Consumer service number recognition using template matching algorithm for improvements in OCR based energy consumption billing," *ICIC Express Lett. Part B Appl.*, vol. 10, no. 10, pp. 895–901, 2019, doi: 10.24507/icicelb.10.10.895.

[14] S. Zhang and Y. Zhou, "Template matching using grey wolf optimizer with lateral inhibition," *Optik (Stuttg).*, vol. 130, pp. 1229–1243, 2017, doi: 10.1016/j.ijleo.2016.11.173.

[15] K. Yogheedha, A. S. A. Nasir, H. Jaafar, and S. M. Mamduh, "Automatic Vehicle License Plate Recognition System Based on Image Processing and Template Matching Approach," *2018 Int. Conf. Comput. Approach Smart Syst. Des. Appl. ICASSDA 2018*, pp. 1–8, 2018, doi: 10.1109/ICASSDA.2018.8477639.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[16] R. Pramanik and S. Bag, “Shape decomposition-based handwritten compound character recognition for Bangla OCR,” *J. Vis. Commun. Image Represent.*, vol. 50, no. November 2017, pp. 123–134, 2018, doi: 10.1016/j.jvcir.2017.11.016.

[17] S. Bansal, M. Gupta, and A. K. Tyagi, “A Necessary Review on Optical Character Recognition (OCR) System for Vehicular Applications,” *Proc. 2nd Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2020*, pp. 918–922, 2020, doi: 10.1109/ICIRCA48905.2020.9183330.

[18] D. Kalina and R. Golovanov, “Application of template matching for optical character recognition,” *Proc. 2019 IEEE Conf. Russ. Young Res. Electr. Electron. Eng. ElConRus 2019*, no. 1, pp. 2213–2217, 2019, doi: 10.1109/EIConRus.2019.8657297.

[19] B. Pan, “Digital image correlation for surface deformation measurement: Historical developments, recent advances and future goals,” *Meas. Sci. Technol.*, vol. 29, no. 8, 2018, doi: 10.1088/1361-6501/aac55b.

[20] C. Moler, “A Brief History of MATLAB,” *MathWorks*, vol. 4, no. June, 2018, [Online]. Available: https://la.mathworks.com/company/newsletters/articles/a-brief-history-of-matlab.html.

[21] A. M. Herrera, H. F. Suhandri, E. Realini, M. Reguzzoni, and M. C. de Lacy, “goGPS: open-source MATLAB software,” *GPS Solut.*, vol. 20, no. 3, pp. 595–603, 2016, doi: 10.1007/s10291-015-0469-x.

[22] M. Öner and I. D. Kocakoç, “JMASM 49: A compilation of some popular goodness of fit tests for normal distribution: Their algorithms and MATLAB codes (MATLAB),” *J. Mod. Appl. Stat. Methods*, vol. 16, no. 2, pp. 547–575, 2017, doi: 10.22237/jmasm/1509496200.

[23] J. Memon, M. Sami, R. A. Khan, and M. Uddin, “Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR),” *IEEE*

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Access, vol. 8, pp. 142642–142668, 2020, doi: 10.1109/ACCESS.2020.3012542.

[24] J. Wang and X. Hu, "Gated recurrent convolution neural network for OCR," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 335–344, 2017.

[25] M. Mieskes and S. Schmunk, "{OCR} Quality and {NLP} Preprocessing," *Proc. 2019 Work. Widening NLP*, no. 2016, pp. 102–105, 2019, [Online]. Available: https://www.aclweb.org/anthology/W19-3633.

[26] L. Hao, Y. Jinsong, T. DIyin, and C. Weizhong, "A real-time can-label monitoring system using OMAP-based OCR," *Proc. 2017 12th IEEE Conf. Ind. Electron. Appl. ICIEA 2017*, vol. 2018-Febru, pp. 1348–1352, 2018, doi: 10.1109/ICIEA.2017.8283048.

[27] S. Malik *et al.*, "An efficient skewed line segmentation technique for cursive script OCR," *Sci. Program.*, vol. 2020, 2020, doi: 10.1155/2020/8866041.

[28] V. Nastase and J. Hitschler, "Correction of OCR word segmentation errors in articles from the ACL collection through neural machine translation methods," *Lr. 2018 - 11th Int. Conf. Lang. Resour. Eval.*, pp. 706–711, 2019.

[29] S. A. Malik, M. Maqsood, F. Aadil, and M. F. Khan, *An efficient segmentation technique for urdu optical character recognizer (OCR)*, vol. 70. Springer International Publishing, 2020.

[30] A. Farhat *et al.*, "OCR based feature extraction and template matching algorithms for Qatari number plate," *2016 Int. Conf. Ind. Informatics Comput. Syst. CIICS 2016*, 2016, doi: 10.1109/ICCSII.2016.7462419.

[31] F. Q. Zhong, P. P. Indurkar, and C. G. Quan, "Three-dimensional digital image correlation with improved efficiency and accuracy," *Meas. J. Int. Meas. Confed.*, vol. 128, no. June, pp. 23–33, 2018, doi: 10.1016/j.measurement.2018.06.022.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[32] S. Elouafiq, "School of Science Engineering HANDWRITTEN DIGITS RECOGNITION : IMAGE CORRELATION VS MACHINE LEARNING EGR 4402 - Capstone Design," 2018.

[33] T. J. Beberniss and D. A. Ehrhardt, "High-speed 3D digital image correlation vibration measurement: Recent advancements and noted limitations," *Mech. Syst. Signal Process.*, vol. 86, pp. 35–48, 2017, doi: 10.1016/j.ymssp.2016.04.014.

[34] G. Chiron, A. Doucet, M. Coustaty, M. Visani, and J. P. Moreux, "Impact of OCR Errors on the Use of Digital Libraries: Towards a Better Access to Information," *Proc. ACM/IEEE Jt. Conf. Digit. Libr.*, 2017, doi: 10.1109/JCDL.2017.7991582.

[35] T. T. H. Nguyen, A. Jatowt, M. Coustaty, N. Van Nguyen, and A. Doucet, "Deep statistical analysis of OCR errors for effective post-OCR processing," *Proc. ACM/IEEE Jt. Conf. Digit. Libr.*, vol. 2019-June, no. 1, pp. 29–38, 2019, doi: 10.1109/JCDL.2019.00015.

[36] M. A. Sutton *et al.*, "Recent Progress in Digital Image Correlation: Background and Developments since the 2013 W M Murray Lecture," *Exp. Mech.*, vol. 57, no. 1, pp. 1–30, 2017, doi: 10.1007/s11340-016-0233-3.

[37] Y. L. Dong and B. Pan, "A Review of Speckle Pattern Fabrication and Assessment for Digital Image Correlation," *Exp. Mech.*, vol. 57, no. 8, pp. 1161–1181, 2017, doi: 10.1007/s11340-017-0283-1.

[38] S. Ghosh, N. Das, I. Das, and U. Maulik, "Understanding deep learning techniques for image segmentation," *ACM Comput. Surv.*, vol. 52, no. 4, 2019, doi: 10.1145/3329784.

[39] L. S. V. Thomas and J. Gehrig, "Multi-Template Matching: A versatile tool for object-localization in microscopy images," *bioRxiv*, vol. 7, pp. 1–8, 2019, doi: 10.1101/619338.

UTeM
اونيورسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[40] P. Xiao, Z. Y. Wu, R. Christenson, and S. Lobo-Aguilar, “Development of video analytics with template matching methods for using camera as sensor and application to highway bridge structural health monitoring,” *J. Civ. Struct. Heal. Monit.*, vol. 10, no. 3, pp. 405–424, 2020, doi: 10.1007/s13349-020-00392-6.

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPENDICES

Appendix A Coding for Optical Character Recognition

```
function OpticalCharacterRecognition
imagen=imread('Sample 4\Lato.png');
figure;
imagen1 = imagen;

imshow(imagen1);
title('Input Image with Noise');

if size(imagen,3)==3
   imagen=rgb2gray(imagen);
   %figure;
   %imagen2=imagen;
   %imshow(imagen2);
   %title('Grayscale Image');
end

threshold=graythresh(imagen);
imagen=~im2bw(imagen,threshold);
imagen=bwareaopen(imagen,2);
figure;
imagen3=imagen;
imshow(imagen3);

%%
RL=imagen;
word=[];
fid=fopen('text.txt','wt');
load templates
global templates
num_letters=size(templates,2);

while 1
    [FL RL]=horizontalSegmentation(RL);
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
     imagen=FL;
     n=0;
     spacevector=[];
     RA=FL;

     while 1
         [FA RA space]=verticalSegmentation(RA);
         imgChar=imresize(FA,[42 24]);
         n=n+1;
         spacevector(n)=space;

         letter=TemplateMatching(imgChar,num_letters);
         word=[word letter];

         if isempty(RA)
             break;
         end
     end

   max_space = max(spacevector);
     no_spaces = 0;

      for x= 1:n
        if spacevector(x+no_spaces)> (0.75 * max_space)
           no_spaces = no_spaces + 1;
             for m = x:n
               word(n+x-m+no_spaces)=word(n+x-m+no_spaces-1);
             end
            word(x+no_spaces) = ' ';
            spacevector = [0 spacevector];
        end
      end

     fprintf(fid,'%s\n',word);
     word=[ ];
     if isempty(RL)
         break
     end
end
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
fclose(fid);
winopen('text.txt')
clear all
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix B Coding for Template Creation

```
function TemplateCreation

A=imread('letters_numbers\A.bmp');B=imread('letters_numbers\B.bmp');
C=imread('letters_numbers\C.bmp');D=imread('letters_numbers\D.bmp');
E=imread('letters_numbers\E.bmp');F=imread('letters_numbers\F.bmp');
G=imread('letters_numbers\G.bmp');H=imread('letters_numbers\H.bmp');
I=imread('letters_numbers\I.bmp');J=imread('letters_numbers\J.bmp');
K=imread('letters_numbers\K.bmp');L=imread('letters_numbers\L.bmp');
M=imread('letters_numbers\M.bmp');N=imread('letters_numbers\N.bmp');
O=imread('letters_numbers\O.bmp');P=imread('letters_numbers\P.bmp');
Q=imread('letters_numbers\Q.bmp');R=imread('letters_numbers\R.bmp');
S=imread('letters_numbers\S.bmp');T=imread('letters_numbers\T.bmp');
U=imread('letters_numbers\U.bmp');V=imread('letters_numbers\V.bmp');
W=imread('letters_numbers\W.bmp');X=imread('letters_numbers\X.bmp');
Y=imread('letters_numbers\Y.bmp');Z=imread('letters_numbers\Z.bmp');

a=imread('letters_numbers\a.png');b=imread('letters_numbers\b.png');
c=imread('letters_numbers\c.png');d=imread('letters_numbers\d.png');
e=imread('letters_numbers\e.png');f=imread('letters_numbers\f.png');
g=imread('letters_numbers\g.png');h=imread('letters_numbers\h.png');
i=imread('letters_numbers\i.png');j=imread('letters_numbers\j.png');
k=imread('letters_numbers\k.png');l=imread('letters_numbers\l.png');
m=imread('letters_numbers\m.png');n=imread('letters_numbers\n.png');
o=imread('letters_numbers\o.png');p=imread('letters_numbers\p.png');
q=imread('letters_numbers\q.png');r=imread('letters_numbers\r.png');
s=imread('letters_numbers\s.png');t=imread('letters_numbers\t.png');
u=imread('letters_numbers\u.png');v=imread('letters_numbers\v.png');
w=imread('letters_numbers\w.png');x=imread('letters_numbers\x.png');
y=imread('letters_numbers\y.png');z=imread('letters_numbers\z.png');

one=imread('letters_numbers\1.bmp');
two=imread('letters_numbers\2.bmp');
three=imread('letters_numbers\3.bmp');four=imread('letters_numbers\4.bm
p');
five=imread('letters_numbers\5.bmp');
six=imread('letters_numbers\6.bmp');
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
seven=imread('letters_numbers\7.bmp');
eight=imread('letters_numbers\8.bmp');
nine=imread('letters_numbers\9.bmp');
zero=imread('letters_numbers\0.bmp');

letter=[A B C D E F G H I J K L M N O P Q R S T U V W X Y Z];
lowercase = [a b c d e f g h i j k l m n o p q r s t u v w x y z];
number=[one two three four five six seven eight nine zero];

character=[letter number lowercase];

N=ones(1,62)*24;
templates=mat2cell(character,42,N);
save('TemplateCreation','templates')
clear all
end
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix C Coding for Template Matching

```
function letter= TemplateMatching(imgChar,num_letters)

global templates
C=[];
for n=1:num_letters
  sem=corr2(templates{1,n},imgChar);
  C=[C sem];
end
vd=find(C==max(C));
if vd==1
    letter='A';
elseif vd==2
    letter='B';
elseif vd==3
    letter='C';
elseif vd==4
    letter='D';
elseif vd==5
    letter='E';
elseif vd==6
    letter='F';
elseif vd==7
    letter='G';
elseif vd==8
    letter='H';
elseif vd==9
    letter='I';
elseif vd==10
    letter='J';
elseif vd==11
    letter='K';
elseif vd==12
    letter='L';
elseif vd==13
    letter='M';
elseif vd==14
    letter='N';
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
elseif vd==15
    letter='O';
elseif vd==16
    letter='P';
elseif vd==17
    letter='Q';
elseif vd==18
    letter='R';
elseif vd==19
    letter='S';
elseif vd==20
    letter='T';
elseif vd==21
    letter='U';
elseif vd==22
    letter='V';
elseif vd==23
    letter='W';
elseif vd==24
    letter='X';
elseif vd==25
    letter='Y';
elseif vd==26
    letter='Z';
    %*-*-*-*-*
elseif vd==27
    letter='1';
elseif vd==28
    letter='2';
elseif vd==29
    letter='3';
elseif vd==30
    letter='4';
elseif vd==31
    letter='5';
elseif vd==32
    letter='6';
elseif vd==33
    letter='7';
elseif vd==34
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
    letter='8';
elseif vd==35
    letter='9';
elseif vd==36
    letter='0';
    %********
elseif vd==37
    letter='a';
elseif vd==38
    letter='b';
elseif vd==39
    letter='c';
elseif vd==40
    letter='d';
elseif vd==41
    letter='e';
elseif vd==42
    letter='f';
elseif vd==43
    letter='g';
elseif vd==44
    letter='h';
elseif vd==45
    letter='i';
elseif vd==46
    letter='j';
elseif vd==47
    letter='k';
elseif vd==48
    letter='l';
elseif vd==49
    letter='m';
elseif vd==50
    letter='n';
elseif vd==51
    letter='o';
elseif vd==52
    letter='p';
elseif vd==53
    letter='q';
```

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
elseif vd==54
    letter='r';
elseif vd==55
    letter='s';
elseif vd==56
    letter='t';
elseif vd==57
    letter='u';
elseif vd==58
    letter='v';
elseif vd==59
    letter='w';
elseif vd==60
    letter='x';
elseif vd==61
    letter='y';
elseif vd==62
    letter='z';
else
    letter='l';

end
end
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix D Coding for Horizontal Segmentation

```
function [FL RL]=horizontalSegmentation(imagen)
imagen=clip(imagen);
m=size(imagen,1);
for s=1:m
    if sum(imagen(s,:))==0
        fl=imagen(1:s-1,:);
        rl=imagen(s:end,:);
        FL=clip(fl);
        RL=clip(rl);

        break;

    else
        FL=imagen;
        RL=[];
    end
end
```

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix E Coding for Vertical Segmentation

```
function [FA RA space]=verticalSegmentation(imagen)
imagen=clip(imagen);
m=size(imagen,2);


for s=1:m
    if sum(imagen(:,s))==0
    fa=imagen(:,1:s-1);
    ra=imagen(:,s:end);
    FA=clip(fa);
    RA=clip(ra);
    space=size(ra,2)-size(RA,2);
    break;
    else
        FA=imagen;
        RA=[];
        space=0;
    end

end
```

اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix F Coding for Clip

```
function image_out=clip(image_in)
[r c]=find(image_in);
image_out=image_in(min(r):max(r),min(c):max(c));
end
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA