Automatic Recognition of Handwritten Score Digits

by

Kang Gim Pin

16270

Dissertation submitted in partial fulfilment of the requirements for the Degree of
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

JANUARY 2016

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Automatic Recognition of Handwritten Score Digits

by

Kang Gim Pin

16270

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

____

Suhaila Binti Badarol Hisham
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

January 2016

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

KANG GIM PIN

ABSTRACT

Despite printed text being widely used since the introduction of computers and printers, several areas such as office automation, e-government, banking and education field still rely on manual data entry. Undeniably, manual data entry is very time consuming and human are prone to make mistakes during this task especially when the amount of the data to be entered is huge. Thus, recognition of handwritten digits plays an important role in life nowadays as it speeds up the data entry process. However, handwritten numerals recognition is a challenging problem as the handwriting styles are varying from person to person. In this project, a handwritten numerals recognition system is developed using Histogram of Oriented Gradients (HOG) as the feature extraction method. Several classifiers were also examined to determine the classifying method with the highest accuracy. The handwriting samples are scanned using an optical scanner and converted into digital images. After that, pre-processing steps such as segmentation, size normalization, and noise removal are applied to the scanned image to facilitate the feature extraction process. The features of each number are then extracted using HOG using cell size of 4 x 4 and the histograms obtained are concatenated into 1-D matrix. Two classifiers, Support Vector Machine (SVM) and k-Nearest Neighbour (kNN) are trained using HOG features obtained to classify the input digit and the result are then tabulated and compared. The result shows that SVM has a higher recognition rate of 83% compared to 67% from kNN classifier. However, the recognition rate is improved to 87% by applying salient structure feature in classification stage after SVM. In addition, a GUI interface is developed to let user to select from 3 assessment form templates to be identified. Digits classified on each template are then exported and recorded in a spreadsheet.

ACKNOWLEDGEMENTS

First of all, I would like to express my sincere gratitude to my supervisor Ms Suhaila binti Badarol Hisham, lecturer of Universiti Teknologi PETRONAS for the continuous support of my final year project, for her patience, motivation, enthusiasm, and immense knowledge. Her guidance helped me in all the time of research and writing of this thesis. I could not have completed my final year project without the guidance from her.

Furthermore, I would like to thank my fellow classmates, colleagues and seniors who willing to spend their times to fill up the score sheet form for my testing data. Their responsive help, encouragements and guidance are the

Subsequently, my sincere thanks also go to my parents for their encouragements and supporting me spiritually throughout my life. Last but not least, superior thanks to everyone who directly or indirectly contributing towards the success of this final year project entitled 'Automatic Recognition of Handwritten Score Digits'.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

HOG	Histogram of Oriented Gradients
SVM	Support Vector Machine
kNN	k-Nearest Neighbours
SST	Score Sheet Template
MNIST	Mixed National Institute of Standards and Technology
GUI	Graphical User Interface
MATLAB	Matric Laboratory
LBP	Linear Binary Pattern
OCR	Optical Character Recognition

CHAPTER 1
INTRODUCTION

1.1 Background of Study

Automatic handwritten recognition is an ability of a system or a program to interpret text written by human. Despite printed text being widely used since the introduction of computers and printers, several important documents in education, finance, ID registration still rely on handwritten input. However, automatic image recognition system are still being developed and refined to identify handwritten images accurately and to relay the information into a database.

Handwritten recognition can be classified into two types, online handwritten recognition and offline handwritten recognition [1]. Online handwritten recognition is a technique that sense the movement of the pen-tip and convert the handwritten text to letter code. The handwritten text is converted as it is written and such technology is being widely used in Personal Digital Assistant (PDA) and smartphone nowadays [1]. On the other hand, offline handwritten recognition takes image as an input and converts the handwritten text contained in the image.

Optical Character Recognition (OCR) is a technology that convert handwritten or printed text into machine-encoded text [2]. The working principle is almost the same with handwritten character recognition. However, the OCR technology is only robust in printed text recognition, but weak in handwritten text recognition due to various people have various styles of handwriting.

In this project, histogram of oriented gradients (HOG) is used as the feature extraction method as it is able to capture edges and gradients structure that is very characteristics of digits. Classifiers such as support vector machines, k-nearest neighbours are trained using HOG features obtained and the performance is evaluated.

1.2 Problem Statement

- Transferring handwritten digits manually takes a lot of time
  When the volume of forms, checks etc. is huge, mistakes can happen due to tiredness, rushed work and parallax error.
- Variations in handwritten character
  Existing optical character recognition (OCR) is well-established for printed numerals or alphabets. However, variations in handwritten character still pose a challenge to OCR systems, with most algorithm reaching 80 - 85% accuracy for handwritten digits.

1.3 Objectives

The objectives of the project are:

1. To develop a program to automatically recognize handwritten numerals from an optically scanned template using one or a combination of several algorithms of character recognition with minimum accuracy of 85%.
2. To develop an interface where user can identify regions within a template that contain required handwritten information.

1.4 Scope of Study

This project was using Mixed National Institute of Standards and Technology (MNIST) database of handwritten digits that has a training set of 60,000 examples, and a test set of 10,000 examples. HOG was used as feature extraction method and two classifiers, SVM and kNN are applied in classification stage. Three templates, Score Sheet (SST), Test Paper and ISDP Final Presentation Assessment were used to collect testing samples. Only the handwritten data of student ID and scores written by students and/or lecturers will be considered for data entry into a database system.

CHAPTER 2
LITERATURE REVIEW

2.1 Feature Extraction Method

2.1.1 Projection Method

Projection method is used since the early stages of OCR. The method reduces two-dimensional information into one-dimension data, thus simplifying the character recognition as a one-dimensional pattern which is easier to process compared to a two-dimensional pattern [3]. The method can be described as having a slit scanning an input character from left to right as shown in Figure 1 (a).

The moving slit projects the light value to the x-axis to reduce the two-dimensional light distribution to one-dimensional light distribution as shown in Figure 1 (b). This method is good and effective for pre-processing in character recognition system such as character segmentation. However, this technique is weak in diagonally oriented patterns, for instance “/” and “\” could not be differentiated, even when a y-axis projection is introduced.

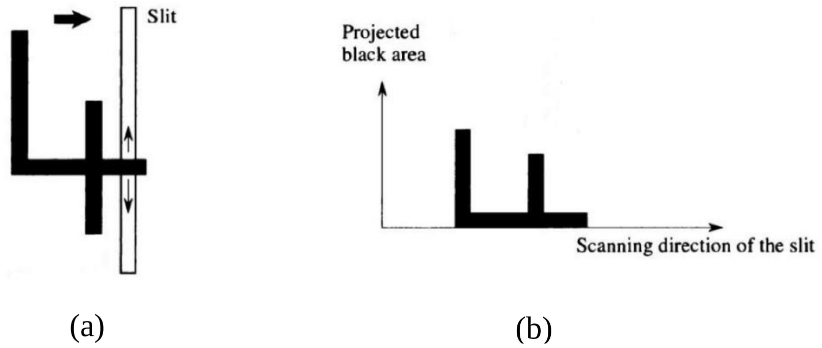


Figure 1: (a) The input character "4" is scanned from left to right (b) The black area is projected onto x-axis in the scanning direction of the slit [3]

2.1.2 Cross-Correlation Method

Cross-correlation method is a simple matching method that can be used to extract feature of a two-dimensional pattern character. The formula used in this method to calculate the similarity of *x* and *y* is shown in (1) [3].

$$S^i(f) = \frac{\iint_R f(x,y)g_i(x,y)dxdy}{\sqrt{\iint_R f(x,y)^2dxdy}\sqrt{\iint_R g_i(x,y)^2dxdy}}\ , \tag{1}$$

where *i* = 1, 2, 3, … , *P*; and *P* is the number of different image patterns of a given alphabet.

The input character *f* is recognized as a given pattern if the maximum value of $S^i(f)$ returns the maximum value. This method works well for processing characters with breaks/gaps and stains/smears. However, this is only valid when the breaks or the stains occur within the stroke of a character, as shown in Figure 2 (a) and 2 (b). In this two cases, the value of the cross-correlation is not much affected. However, as shown in Figure 2 (c) and 2 (d), if there is a break or stain at the boundary that opens out of a character’s domain, this noise would be considered significant by the method thus the output is not reliable.

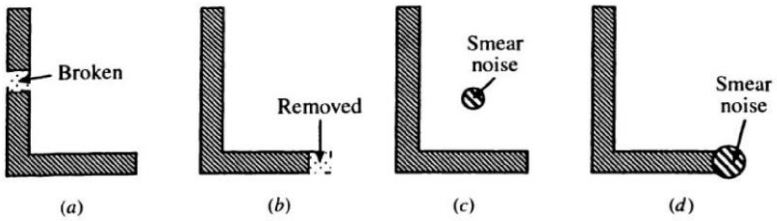


Figure 2: Illustration of cross-correlation method against negative noise (a and b) and positive noise (c and d) [3]

2.1.3 Logical Matching

Logical matching, also known as matrix matching, is another method that is used in character recognition. The principle of this method is shown in Figure 3, in which an input character “A” is successfully identified. The result is based on the logical decision that all the pixel points B_1 to B_4 lying on the stroke have value 1 and all the pixels points W_1 to W_4 lying on the background have value of 0. In this case white colour carries value 0 while black colour carries value 1.

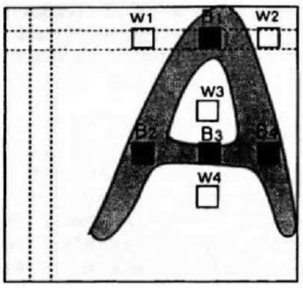


Figure 3: Illustration of logical matching [3]

However, this technique is very rigid as one local change at the detecting points would affect the decision decisively. For example, suppose that the part of W_3 is changed to 1, then the logical value of “A” changes to 0 as it does not match against the template of “A”. Although several ways have been introduced to relax the rigid logical expression, this technique is still weak in position displacement.

2.1.4 Crossing Method

Most numerals can be generally constructed out of strokes drawn around two focal points, such as 2, 3, 5, 6, 8 and 9. Sonde method uses this concept to recognize numerals. In this method, two points are set in a frame, marked by circles. Then four detecting bars are radially arranged in different directions: top, bottom, right and left. If an input numeral crosses the bars, then “1” is counted, and otherwise “0” is counted. These counts are then listed in a form of vector for each input numeral. For instance in Figure 4, the result of numeral 3 is (1, 1, 1, 1, 0, 0). This technique is robust against position displacement, however, it is weak in size variation.

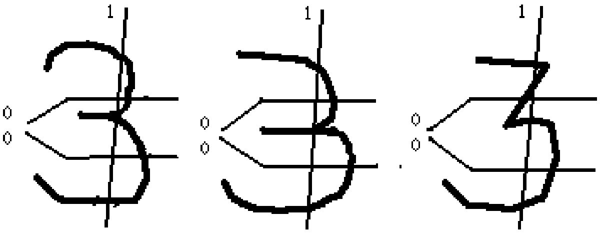


Figure 4: Illustration of Sonde method for different styles of numerals 3 [4]

2.1.5 Geometrical Features Method

Every numeral has its geometrical features such as end points, crossing points and branching points. These features are easy to be extracted and expected to be robust against variations of in handwritten characters. However, this feature is not complete as the number of terminals, branching points and crossing points cannot differentiate numbers within the numeral group of (1, 2, 7) and (6, 9). Thus, another geometrical feature is added to resolve this issue, which is the direction at the end point. The direction is quantized into eight directions as shown in Figure 5 (a). For handwritten characters, there will be variations in the directions, thus some end point directions are defined in specific range, for example, [3, 5] means the direction lies within 3 through 5 as shown in Figure 5 (a).

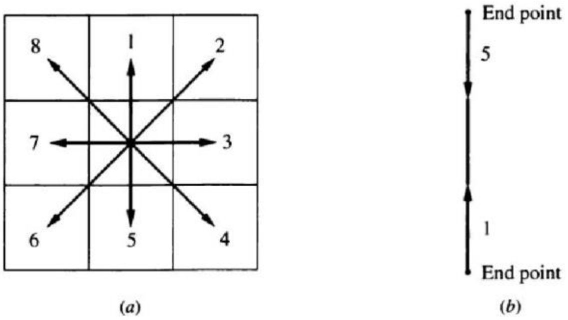


Figure 5: (a) Eight quantized directions; (b) Illustration of the direction at an end point [3]

In addition, another feature is introduced to make this method more robust, which is the position of the end point directional feature. By introducing an appropriate coordinate system, end points are projected to y-axis, and the scalar value on the y-axis is taken as the feature. For instance, the number “2” has two end points (T), which corresponds to directional features of [1, 3] and 7. Since the end point of [1, 3] (direction) is located higher than the end point of 7 (direction) in “2”, the positional relationship can be expressed as [1, 3] > 7. Table 1 shows the list of geometric features of handwritten numerals (0 to 9).

Table 1: Geometric Features of standard Handwritten Numerals [3]

Class	Terminals	Branching Points	Crossing Points	Direction	Positional Relationship
0	0				
1	2			1, 5	
2	2			[1, 3], 7	[1, 3] > 7
3	3(2)	1(0)		[1, 3], [3, 5]	
4	4		1		
5	3(2)	1(0)		[3, 5], 7	[3, 5] < 7
6	1	1		[5, 8]	
7	2			1, 1	
8	0		1		
9	1	1		[1, 4]	

2.1.6 Local Binary Pattern

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighbourhood of each pixel and considers the result as a binary number [5]. LBP histogram is used to represents the grayscale image by converting it into array as shown in Figure 6. It has been used in many applications such as motion analysis, face expression recognition and biomedical image analysis [6].

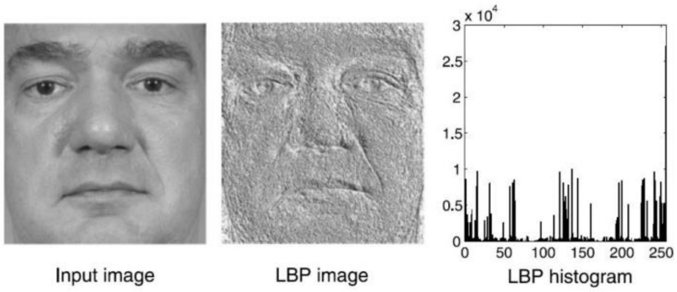
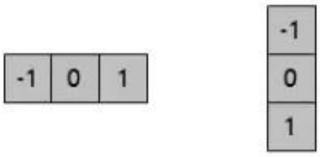


Figure 6: Example of LBP image and histogram for a given input image [6]

2.1.7 Histogram of Oriented Gradients (HOG)

Histogram of Oriented Features was firstly introduced by Navneet Dalal and Bill Triggs for the purpose of object detection [7]. The main concept of this method is that the local object appearance or shape can be described by counting the occurrences of gradient orientation in localized portions of an image. First, the image will be divided into small connected region, called cells. Then, the centered horizontal and vertical derivatives are computed by convolving the image with the filter kernels as shown in Figure 7.



(a) (b)

Figure 7: Filter kernels for (a) x-direction and (b) y-direction

Once the x-derivatives, s_x^2 and y-derivatives, s_y^2 are obtained, the gradient magnitude, s is computed using (2).

$$s = \sqrt{s_x^2 + s_y^2} \tag{2}$$

In addition, gradient orientation also can be calculated from the x and y derivatives obtained using (3).

$$\theta = \arctan(\frac{s_y}{s_x}) \tag{3}$$

In normal practice, the gradient orientation is quantized into 9 bins (0 – 180 degrees) as shown in Figure 8.

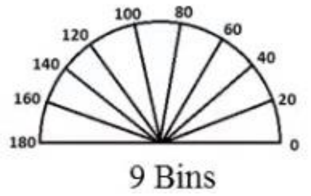


Figure 8: The gradient orientation is quantized into 9 bins

The vote of the orientation histogram comes from the gradient magnitude. For example, in one cell the gradient magnitude computed is 2 and the orientation is 60 degrees, thus the vote of 60 degrees histogram will be increased by 2. In short, the gradient direction of the pixels in each cell will be determined and computed in a histogram form as shown in Figure 9. The HOG descriptor is the concatenation of all these histograms. Last but not least, the local histograms can be contrast normalized to improve the accuracy [8].

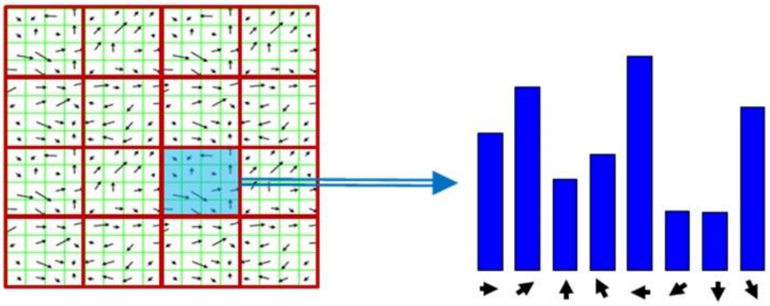


Figure 9: Illustration of HOG descriptor [8].

The summary of the methods discussed in the previous section is shown in Table 2.

Table 2: Summary of various feature extraction method.

Methods	Strengths	Weakness
Projection Method [3]	• Easy to handle in one dimensional	• Could not differentiate diagonally oriented pattern.
Cross-Correlation Method [3]	• Robust against breaks/gaps and stains/smears. (Only true while it occurs within the stroke)	• Weak in position displacement.
Logical Matching [3]	• Simple and easy to be applied.	• Too rigid. • Weak in position displacement.
Crossing Method / Sonde Method [3]	• Robust in position displacement • Good in variations in character shape	• Unpredictable. • Weak in variation in size.
Geometrical Features Method [3]	• Robust in variations in character size • Robust in position displacement	• Weak in variation in character shape.
Local Binary Pattern [6]	• Can be applied in various application	• The histogram of all digits have almost similar pattern. • More suitable to be used in texture based.
Histogram of Oriented Gradients [7]	• It captures edge or gradient structure that is very characteristic of local shape	• Will generate a big number of vector feature.

2.2 Classifiers

2.2.1 Artificial Neural Network

Artificial Neural Network is one of the popular classifiers in handwritten recognition nowadays. Neural network can produce high recognition rate as it has self-learning, self-adapt and self-process capabilities which make it good for handwritten recognition system [9]. There are three layers in neural network, which are the input layer, output layer and the hidden layer. These three layers communicates with each other to train training sets for recognition algorithms. On top of that, it can be trained to perform error-correction learning rule. From the literature, it is reported that neural network can achieve recognition accuracy of more than 90% with appropriate learning rate (μ) being chosen [9].

2.2.2 Support Vector Machine (SVM)

SVM is a discriminative classifier formally defined by a separating hyperplane. Let $(x_i, y_i)_{1\leq i\leq N}$ be a set of training examples, each example $x_i \in R^d$, belongs to a class labelled by $y_i \in \{-1,1\}$, where d is the dimension of input space. The aim of SVM is to define a hyperplane which divides the set of examples such that all points with the same label are on the same side of the hyperplane. The set of samples are considered as linearly separable if (4) is satisfied [10]

$$y_i(w.x_i + b) > 0, \quad i = 1, \dots, N \tag{4}$$

where, *w* is the normal vector to the hyperplane and *b* is the centre point of the division. The best hyperplane would leave the maximum margins from both classes. For example in Figure 10, z_2 is bigger than z_1. Hence we can conclude that the green hyperplane is better than the red hyperplane.

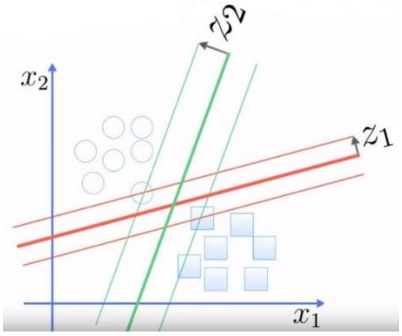


Figure 10: Two variables, x_1 and x_2 are classified by the two hyperplane, z_1 and z_2.

2.2.3 k-Nearest Neighbours

K nearest neighbours is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). Euclidean distance between the test sample and all the training samples will be calculated by using equation (5) [11].

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 + y_2)^2} \quad (5)$$

where *D* is the distance between test sample and training sample and *k* is the number of samples that are nearest neighbours to the test sample. For the example in Figure 11, let *k* = 3. The 3 nearest elements to the test sample (green dot) are 2 “red triangles” and 1 “blue square”. Hence, conclusion can be made that the test sample falls in the same class with “red triangle”. However, if *k* is set to be 5, then the test sample will be classified as “blue square” as the 5 nearest elements to the test samples are 3 “blue squares” and 2 “red triangles”. Thus, selection of *k* value is the most crucial part in this method as it would affect the accuracy directly.

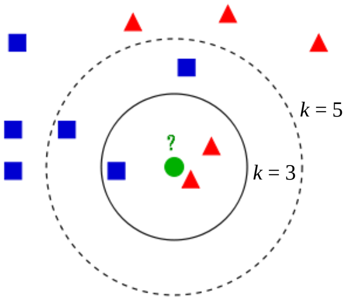


Figure 11: Classification using kNN method with *k* = 3 and *k* = 5

2.2.4 Chi Square Distance

For small sample sizes, chi square distance performs better than other classifier methods due to its stability [12]. The dissimilarity between testing and training images can be calculated using chi square distance as shown in equation (6)

$$\chi^2(\mathrm{S,M}) = \sum_{b=1}^{B} \frac{(\mathrm{S}_b - \mathrm{M}_b)^2}{\mathrm{S}_b + \mathrm{M}_b} \tag{6}$$

where B is the number of features, $\mathcal{S}_b$ is the test sample and M_b is the training sample at b^{th} feature. The similarity between a sample and a model is high when the value of χ^2 is reduced.

CHAPTER 3
METHODOLOGY

3.1 Project Activities

The flow of the project activities is shown in Figure 12. First, literature review is carried out to study various techniques used in digits recognition. Then, select the most suitable feature extraction method and classifier to be used in this project. Next, template is ready to collect the project samples.

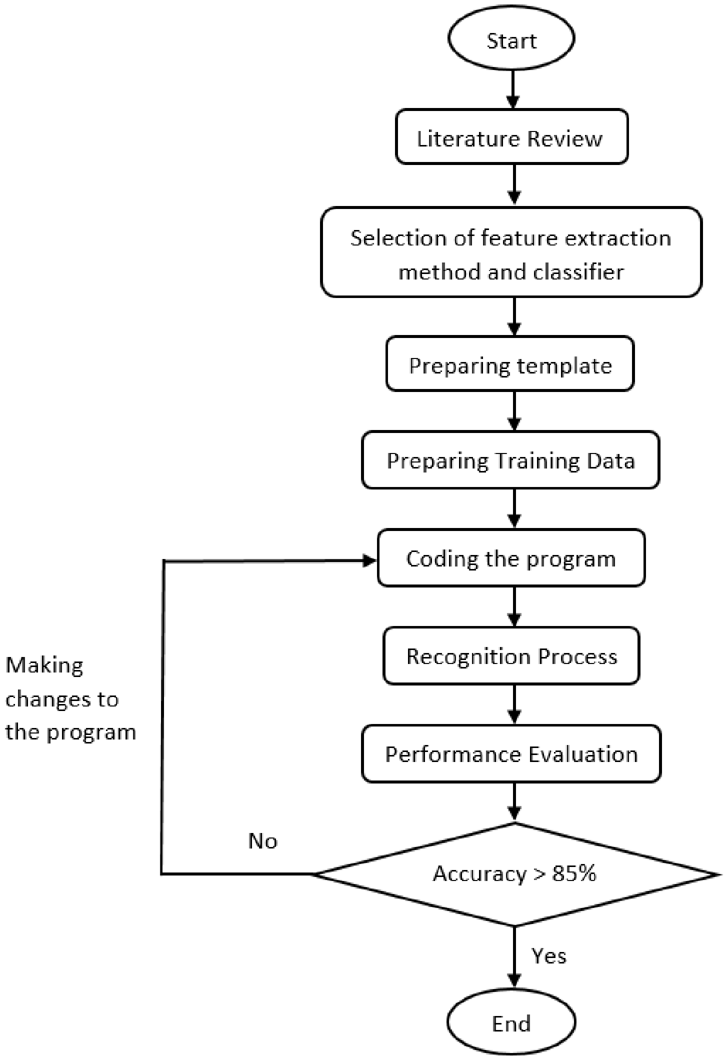


Figure 12: Project activities flow

Coding of the program is started once the training data is ready. After that, recognition process is carried out and the performance is evaluated. The process of recognition will be discussed in more details next section. If the accuracy cannot achieve a recognition rate of at least 85%, some changes shall be made to the program to improve the accuracy.

3.2 Handwritten Digit Scores Recognition Process

The process of offline handwritten recognition is shown in Figure 13. First, the template is scanned using an optical scanner and save as Joint Photographic Experts Group (JPEG) format. The quality of the image scanned should not below 300 Dots per Inch (DPI) as it might affect the recognition rate.

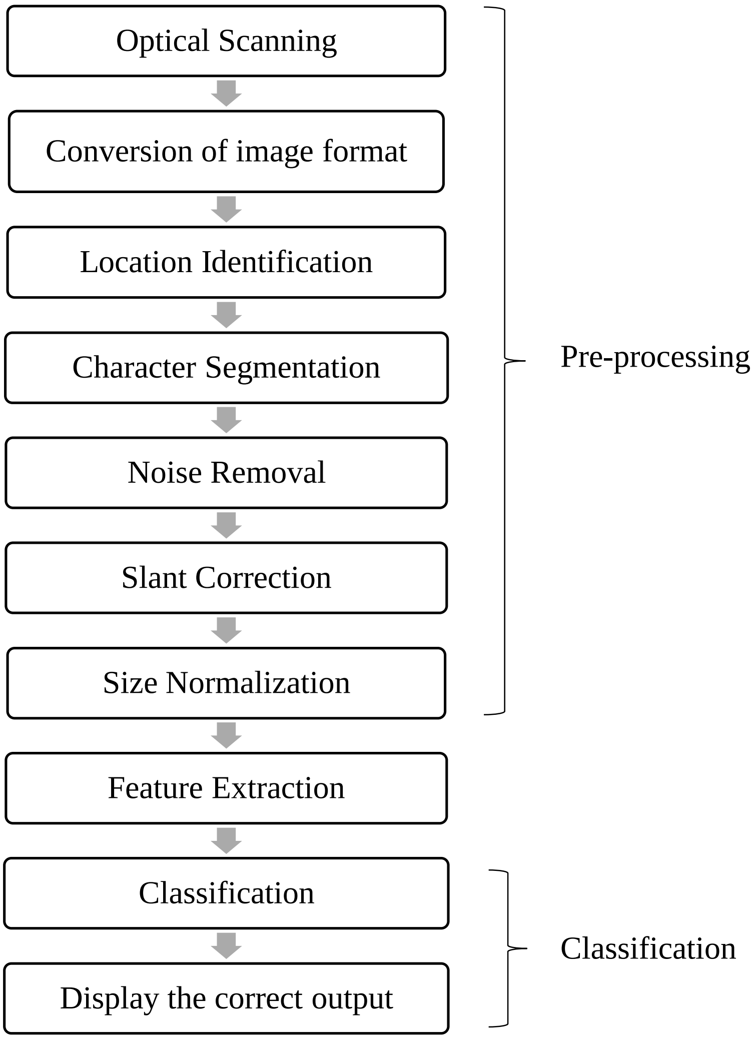


Figure 13: Handwritten Digit Recognition Process (Offline)

The scanned image is then transferred to MATLAB programming code to continue the process. Next, the location of interest is identified and cropped. The handwritten digits is then separated individually by applying segmentation process to the cropped image. However, there is a limitation in this process, connected numbers could not be segmented as connected numbers are identified as a single image.

After that, noise removal is carried out to remove those irrelevant noise and following by slant correction. The image is then go through size normalization process. Next, the pre-processed image will be go through feature extraction process in order to obtain the attributes of patterns that are most pertinent to the classification task. Finally, once the pattern is mapped onto points in feature space, the classification is applied to assign each point in the space to the defined classes [13].

In this project, SVM classifier is used to predict the class label based on the training data. The output of SVM comes from the highest posterior probability of a class. Higher posterior probability scores indicates higher similarity score between the sample and the training data. The outputs (0 to 9) are scaled using their posterior probability scores from high to low. The top 3 outputs are then verified with the salient structural features of the digit till it match. If the top 3 outputs cannot match with any structural features of all the digits, the sample will be classified as the one with highest posterior probability score.

3.3 Salient Structural Feature

Circle is one of the important salient structural features of digits [5]. This can be seen as digit ‘8’ has two circles; digit ‘0’ has a big circle; digits ‘6’ and ‘9’ have one circle but with different position (top circle and bottom circle). Location of nodes is used to calculate circle as shown in Figure 14. If the right side pixels between node 1 and node 2 contain binary value of ‘1’ to a certain threshold, then it is considered as closed loop. This concept also applies to the left side pixels between node 1 and node 2. The digit is assumed to contain a circle when both left and right sides are closed loops. Same concept is applied to node 2 and node3. For example, numeral ‘3’ consist of 3 nodes. For interval of node 1 and node 2, there is a closed loop on the right hand side and open loop on the left hand side. The same goes to node 2 and node 3. If these conditions are fulfil it can be safely assumed that it is a numeral ‘3’.

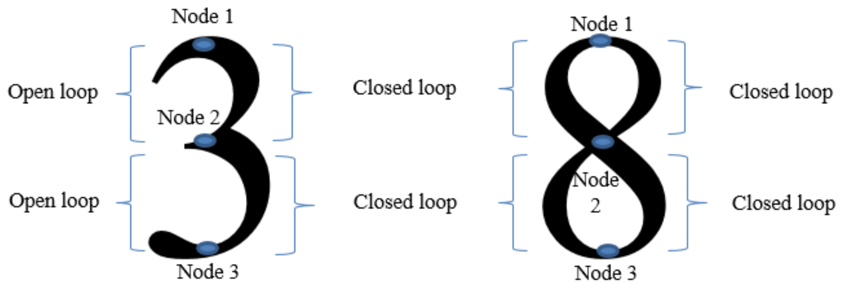


Figure 14: The concept of closed loop and open loop for circle detection[6]

On top of that, numeral '1' usually has much height compared to its width, so the ratio of the height over the width is used to classify whether the test sample is a digit '1'. Numeral '4' and '7' is differentiate by the position of horizontal line. While digit '8' must consist of 2 circles. All these unique salient structure features for numerals 0 – 9 are summarized in Table 3 below.

Table 3: Salient structural features of digits [6]

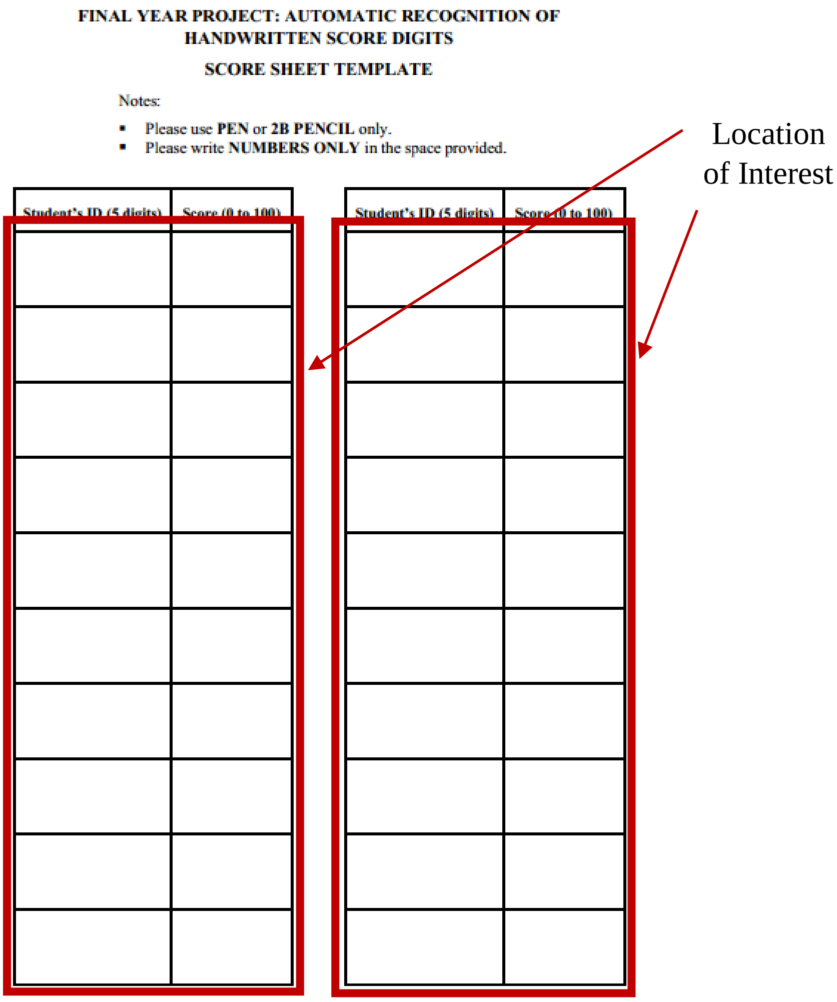
Digit	Salient Structural Features
0	• 2 nodes intersect vertically in the middle • One big circle
1	• The ratio of height over width must be greater than 3
2	• Interval of node 1 and node 2 ✓ Closed loop for right side and open loop for left side • Interval of node 2 and node 3 ✓ Open loop for right side and closed loop for left side
3	• Interval of node 1 and node 2 ✓ Closed loop for right side and open loop for left side • Interval of node 2 and node 3 ✓ Closed loop for right side and open loop for left side
4	• The position of horizontal straight line
5	• Interval of node 1 and node 2 ✓ Open loop for right side and closed loop for left side • Interval of node 2 and node 3 ✓ Closed loop for right side and open loop for left side
6	• Interval of node 1 and node 2 ✓ Open loop for right side and closed loop for left side • Bottom circle
7	• The position of horizontal straight line
8	• 3 nodes intersect vertically in the middle • Top circle and bottom circle

9	• Top circle • Interval of node 2 and node 3 ✓ Closed loop for right side and open loop for left side

3.4 Templates

There are 3 templates used in this project, Score Sheet Template (SST), Test Paper, and ISDP Final Presentation Assessment.

3.4.1 Score Sheet Template (SST)



FINAL YEAR PROJECT: AUTOMATIC RECOGNITION OF HANDWRITTEN SCORE DIGITS

SCORE SHEET TEMPLATE

Notes:

- Please use **PEN** or **2B PENCIL** only.
- Please write **NUMBERS ONLY** in the space provided.

Student's ID (5 digits)	Score (0 to 100)

Student's ID (5 digits)	Score (0 to 100)

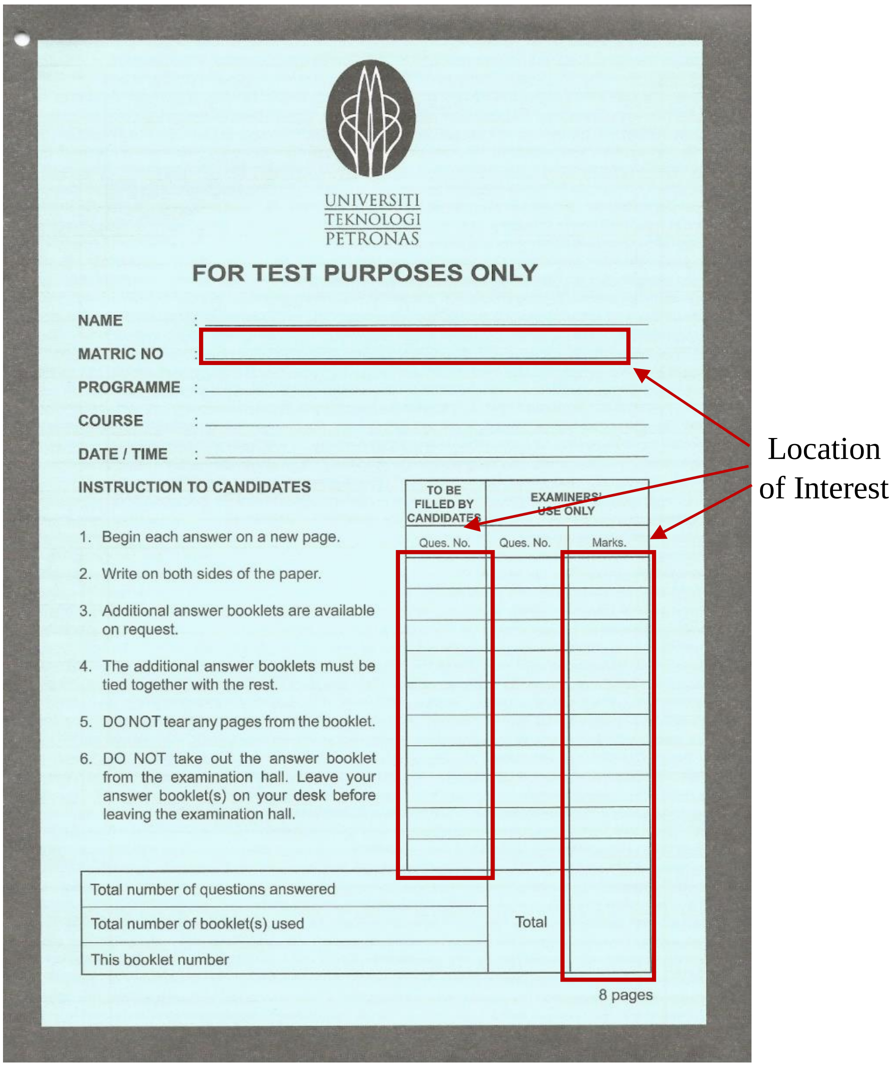
Location of Interest

Figure 15: Location of interest for Score Sheet Template

A Score Sheet Template (SST) is designed for this project as shown in Figure 15. There are 20 spaces available in each sheet to fill in the student ID and scores. Student

ID is limited to 5 digits only and the score range is from 0 to 100. Thus, there are total of about 140 handwritten digits in each SST. SSTs that already filled in by students are collected to test the system accuracy. The result obtained will be discussed in next section. Total of 15 samples have been collected for this template.

3.4.2 Test Paper



UNIVERSITI TEKNOLOGI PETRONAS

FOR TEST PURPOSES ONLY

NAME :

MATRIC NO :

PROGRAMME :

COURSE :

DATE / TIME :

INSTRUCTION TO CANDIDATES

1. Begin each answer on a new page.
2. Write on both sides of the paper.
3. Additional answer booklets are available on request.
4. The additional answer booklets must be tied together with the rest.
5. DO NOT tear any pages from the booklet.
6. DO NOT take out the answer booklet from the examination hall. Leave your answer booklet(s) on your desk before leaving the examination hall.

TO BE FILLED BY CANDIDATES	EXAMINERS' USE ONLY	
Ques. No.	Ques. No.	Marks.

Total number of questions answered	Total
Total number of booklet(s) used	
This booklet number	

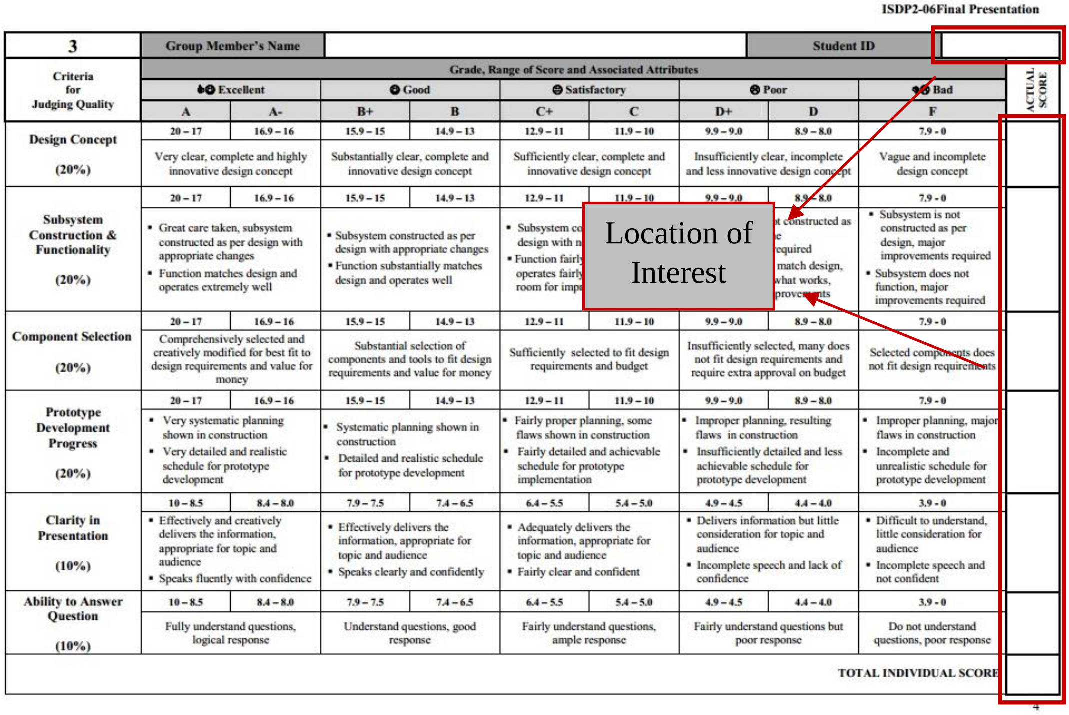
8 pages

Location of Interest

Figure 16: Location of interest for Test Paper

The Test Paper template is the front page of UTP Test answer booklet and has been widely used in UTP during test. Student ID, question number and the marks are the numbers to be identified in this template. 11 samples of Test Paper that contain handwritten information have been collected for testing the system accuracy.

3.4.3 ISDP Final Presentation Assessment



ISDP2-06Final Presentation

3	Group Member's Name							Student ID		
Criteria for Judging Quality	Grade, Range of Score and Associated Attributes									ACTUAL SCORE
	Excellent		Good		Satisfactory		Poor		Bad	
	A	A-	B+	B	C+	C	D+	D	F	
Design Concept (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	Very clear, complete and highly innovative design concept		Substantially clear, complete and innovative design concept		Sufficiently clear, complete and innovative design concept		Insufficiently clear, incomplete and less innovative design concept		Vague and incomplete design concept	
Subsystem Construction & Functionality (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	▪ Great care taken, subsystem constructed as per design with appropriate changes ▪ Function matches design and operates extremely well		▪ Subsystem constructed as per design with appropriate changes ▪ Function substantially matches design and operates well		▪ Subsystem co design with n ▪ Function fairly operates fairly room for impr		t constructed as e equired match design, what works, provements		▪ Subsystem is not constructed as per design, major improvements required ▪ Subsystem does not function, major improvements required	
Component Selection (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	Comprehensively selected and creatively modified for best fit to design requirements and value for money		Substantial selection of components and tools to fit design requirements and value for money		Sufficiently selected to fit design requirements and budget		Insufficiently selected, many does not fit design requirements and require extra approval on budget		Selected components does not fit design requirements	
Prototype Development Progress (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	▪ Very systematic planning shown in construction ▪ Very detailed and realistic schedule for prototype development		▪ Systematic planning shown in construction ▪ Detailed and realistic schedule for prototype development		▪ Fairly proper planning, some flaws shown in construction ▪ Fairly detailed and achievable schedule for prototype implementation		▪ Improper planning, resulting flaws in construction ▪ Insufficiently detailed and less achievable schedule for prototype development		▪ Improper planning, major flaws in construction ▪ Incomplete and unrealistic schedule for prototype development	
Clarity in Presentation (10%)	10 – 8.5	8.4 – 8.0	7.9 – 7.5	7.4 – 6.5	6.4 – 5.5	5.4 – 5.0	4.9 – 4.5	4.4 – 4.0	3.9 - 0	
	▪ Effectively and creatively delivers the information, appropriate for topic and audience ▪ Speaks fluently with confidence		▪ Effectively delivers the information, appropriate for topic and audience ▪ Speaks clearly and confidently		▪ Adequately delivers the information, appropriate for topic and audience ▪ Fairly clear and confident		▪ Delivers information but little consideration for topic and audience ▪ Incomplete speech and lack of confidence		▪ Difficult to understand, little consideration for audience ▪ Incomplete speech and not confident	
Ability to Answer Question (10%)	10 – 8.5	8.4 – 8.0	7.9 – 7.5	7.4 – 6.5	6.4 – 5.5	5.4 – 5.0	4.9 – 4.5	4.4 – 4.0	3.9 - 0	
	Fully understand questions, logical response		Understand questions, good response		Fairly understand questions, ample response		Fairly understand questions but poor response		Do not understand questions, poor response	
									TOTAL INDIVIDUAL SCORE	

4

Figure 17: Location of interest for ISDP Final Presentation Assessment

Assessment form shown in figure above is used when lecturers doing evaluation for students in Integrated System Design Project (ISDP) Final Presentation. The area that contain Student ID and scores is the location of interest in this template. 27 samples have been collected to test the system accuracy and the result is discussed in next section.

3.5 Graphical User Interface (GUI)

A GUI is developed to have three major functions:

1) Enable user to select from the 3 assessment form templates available to be read. User can choose the type of template (SST, Test Paper, ISDP Final Presentation Assessment) and select which sample to be read.
2) Identify the handwritten information that required by the user. User can enter which line of number to be identified.
3) Export all the digits classified and stored in a spreadsheet. User can export the identified digits and record in spreadsheet.

3.6 Software and Hardware Tools

Software

- MATLAB R2015a.
  The program coding and Graphical User Interface (GUI) of this project are done using MATLAB R2015a.

Hardware

- Personal Computer
- Optical Scanner

Details/Week	FYP 1														FYP 2													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Literature Review (Feature extraction and classification methods)	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦																				
Selection of feature extraction method									■																			
Preparing templates									🟦																			
Collection the project samples										🟦	🟦	🟦	🟦															
Developing the program coding (Pre-processing)											🟦	🟦																
Pre-processing coding is ready												■	◆															
Preparing Training data													🟦	🟦														
Continue developing program coding (HOG feature extraction) and testing													🟦	🟦	🟦	🟦	🟦											
Feature extraction coding ready																	■	◆										
Developing coding for classifiers																		🟦	🟦	🟦	🟦	🟦						
Coding for classifiers is ready. Test the classifiers and compare the accuracy																						■	◆					
Develop GUI interface																						🟦	🟦	🟦				
GUI Interface ready																								■	◆			

🟦 Project Activities ■ Key Milestone / Project Achievements ◆ Achieved

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Pre-Processing

As mentioned in the previous section, the collected samples will be scanned using an optical scanner and save in Joint Photographic Experts Group (JPEG) format. The scanner used in this project is 300 dots per inch (DPI) resolution. The quality of scanned image should not be poor as it might affects the performance of recognition process.

The scanned image is then fetched into MATLAB programming coding manually. The image will first transformed to binary image and inverted which consist of black and white colour only. Each column that contain student ID and scores is cropped as shown in Figure 18.

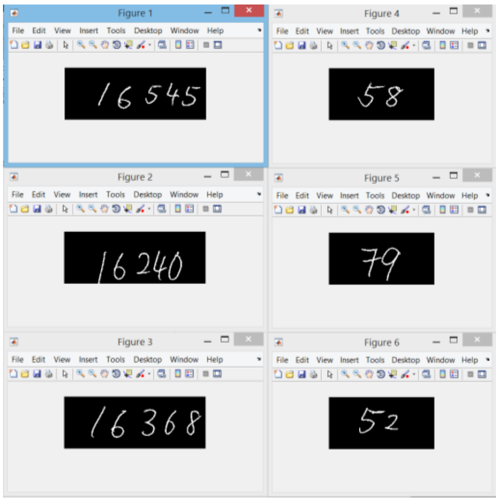
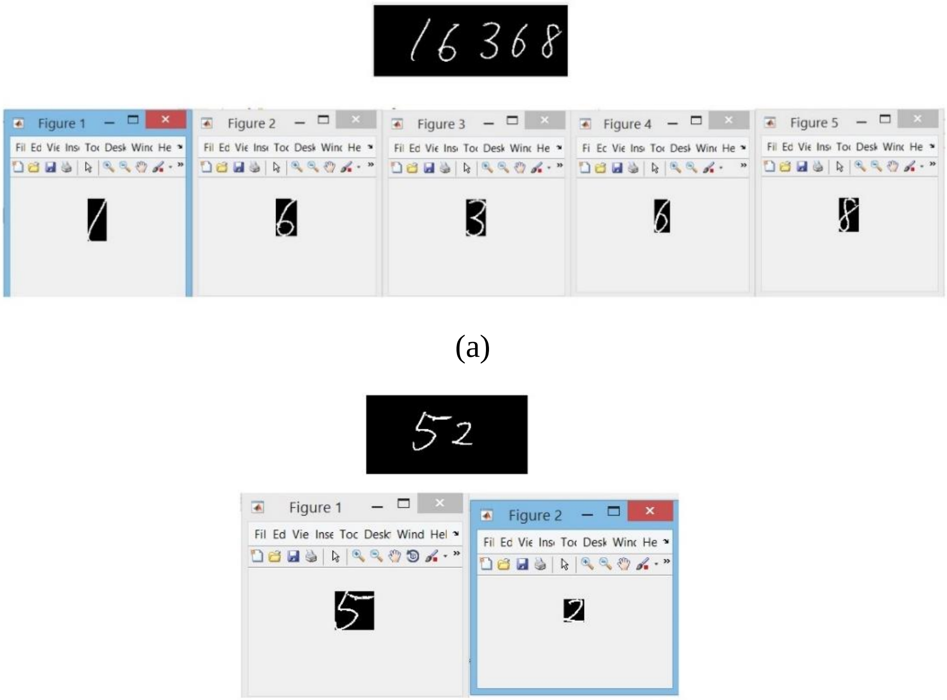


Figure 1
File Edit View Insert Tools Desktop Window Help
16545
Figure 4
File Edit View Insert Tools Desktop Window Help
58
Figure 2
File Edit View Insert Tools Desktop Window Help
16240
Figure 5
File Edit View Insert Tools Desktop Window Help
79
Figure 3
File Edit View Insert Tools Desktop Window Help
16368
Figure 6
File Edit View Insert Tools Desktop Window Help
52

Figure 18: Result of cropped image

The handwritten digits in the cropped image is then segmented using the function “regionprops” which can be found in Image Processing Toolbox in MATLAB. The result of segmentation process is shown in Figure 19 (a) and (b). However, the weakness of this method is that it cannot segment connected numbers because such connected numbers will be treated as a single image.



(a)

(b)

Figure 19 (a) and (b): The output of segmentation process

Next, noise removal is carried out to remove unwanted noise that may affect the output accuracy. The suitable threshold value of pixels should be chosen as it may lead to removal of some important parts of the number that we want to recognize. On top of that, size normalization is then applied on these segmented numbers and the result is shown in Figure 20. This step is to make sure the features length extracted from the image are consistent.

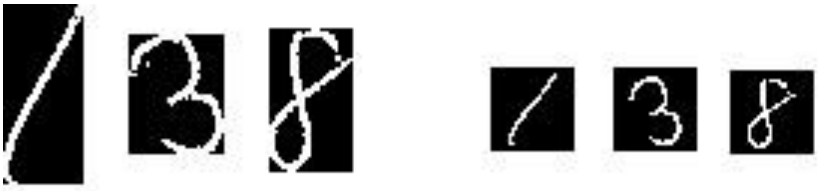


Figure 20: Example of numbers ‘1’, ‘3’, and ‘8’ go through size normalization process.

4.2 Features from Histograms of Oriented Gradients (HOG)

The features dimension obtained by the HOG are different depending on the cell size, as shown in Figure 21. Smaller cell size can capture more information but at the cost of longer processing time. Larger cell size only capture large scale spatial information and thus the feature dimension is smaller compared to smaller cell size.

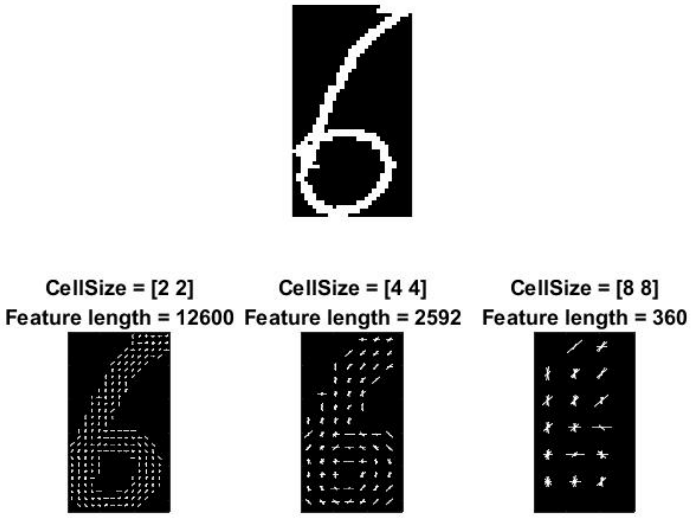


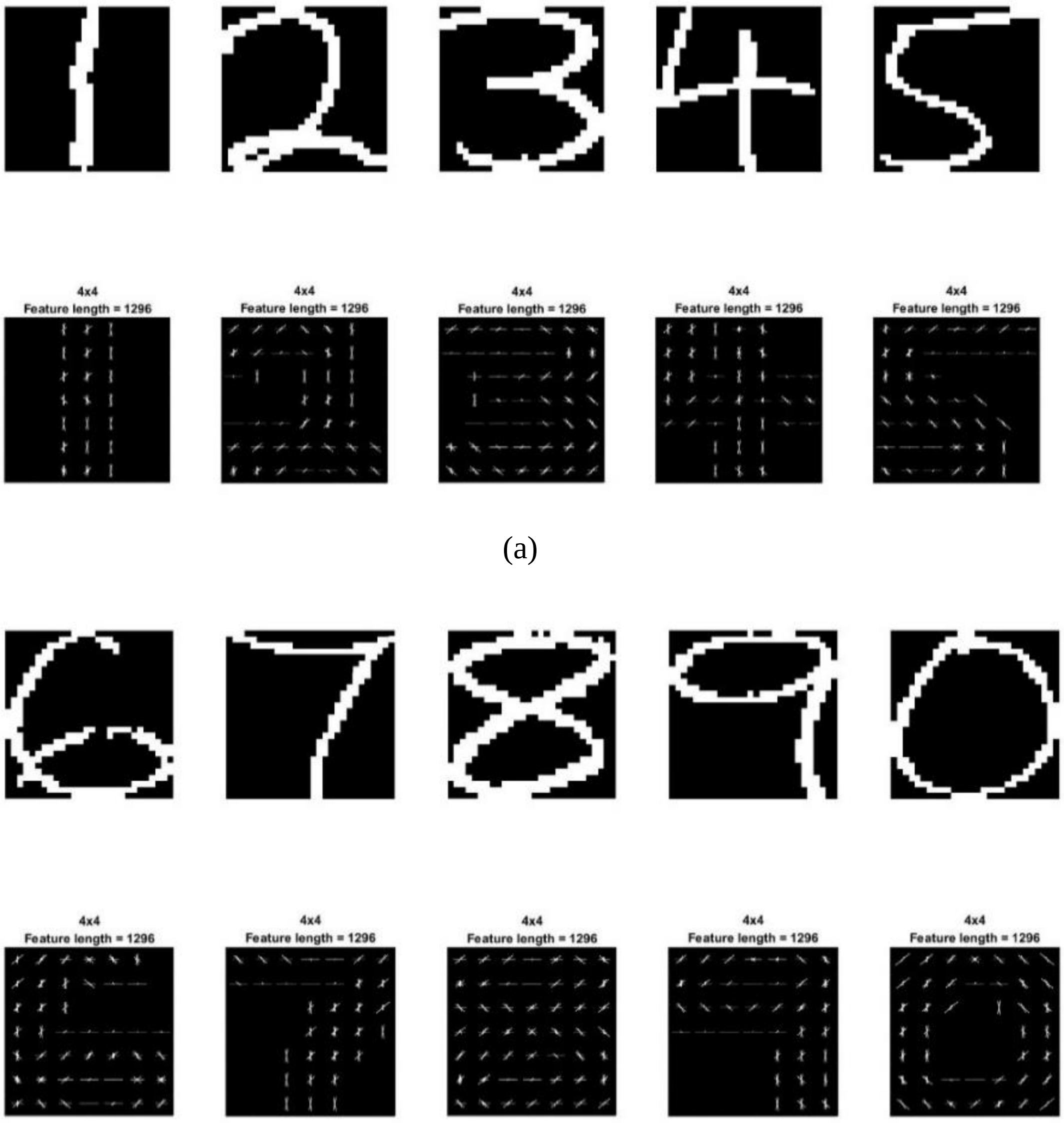
Figure 21: Features captured using HOG with different cell size.

From the figure above, cell size of 2 x 2 can capture most information (Feature length = 12600) compared with cell size of 4 x 4 and 8 x 8. However, cell size of 4 x 4 is the most suitable for handwritten recognition as the information captured is sufficient and the feature dimension is not too large to burden the training time.

In this project, cell size of 4 x 4 is used. The 28 x 28 pre-processed image is divided into 8 x 8 blocks with 50% overlap. Therefore there will be total of 36 blocks. Each block consist of 2 x 2 cells with size of 4 x 4. The feature dimension of the image can be calculated as below:

$$\begin{aligned} Feature\ Dimension &= No\ of\ blocks\ \times No\ of\ cells\ \times Quantized\ Orientation \\ &= 36\ \times\ 4\ \times 9 \\ &=\ 1296 \end{aligned}$$

The visualization of features extracted for handwritten numbers 0 to 9 using HOG is shown as figure below.



(a)

(b)

Figure 22(a) & (b): The visualization of features extracted for sample numbers 0-9

Figure 22 above shows that HOG descriptor manage to capture most of the important characteristic of the numbers using cell size of 4 x 4.

The training data used is obtained from the MNIST database [14]. The database can be divided into two parts, training database and testing database. There are 60,000 samples in training database and 10,000 samples in testing database. Several testing has been carried out by training SVM classifier with different numbers of training data.

The testing is focus on one of the sample image that using Score Sheet Template (SST) and the result is shown in table below.

Table 4: Recognition rate based on different numbers of training data used

From MNIST Training Database

From MNIST Testing Database

Number of Training Data	Training time	Accuracy
10,000	92 seconds	76.4%
20,000	149 seconds	75.7%
30,000	369 seconds	80.7%
40,000	417 seconds	75.7%
60,000	648 seconds	77.9%
10,000	88 seconds	80.0%

As observed from Table 4, the recognition rate does not improved when the number of training data increased, but it does lengthen the training time. The performance of using 10,000 samples from testing database as training data is the best among others. (fastest training time & high accuracy) Hence, 10,000 samples from MNIST testing database is used as a training data to train SVM classifier in this project. There are around 1000 samples of training data are used for each digit. Examples of MNIST testing database that used as training data for numeral '0' is shown in Figure 23.

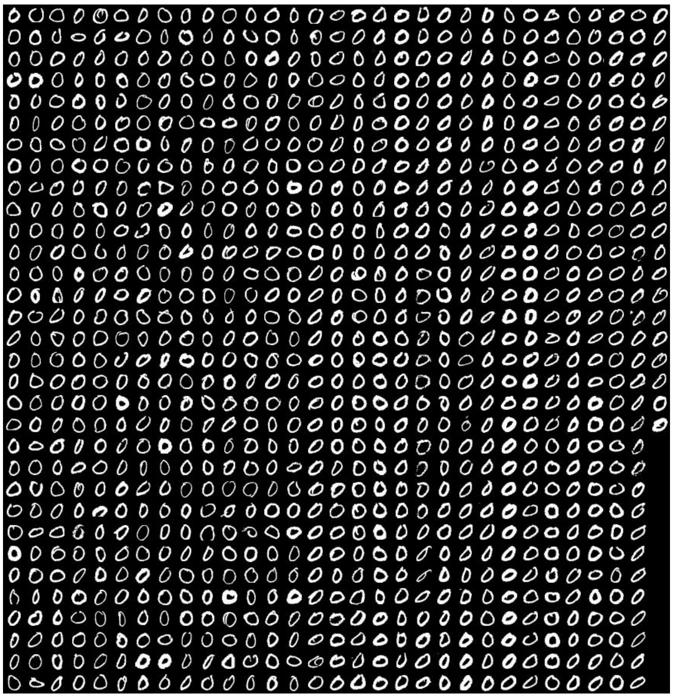
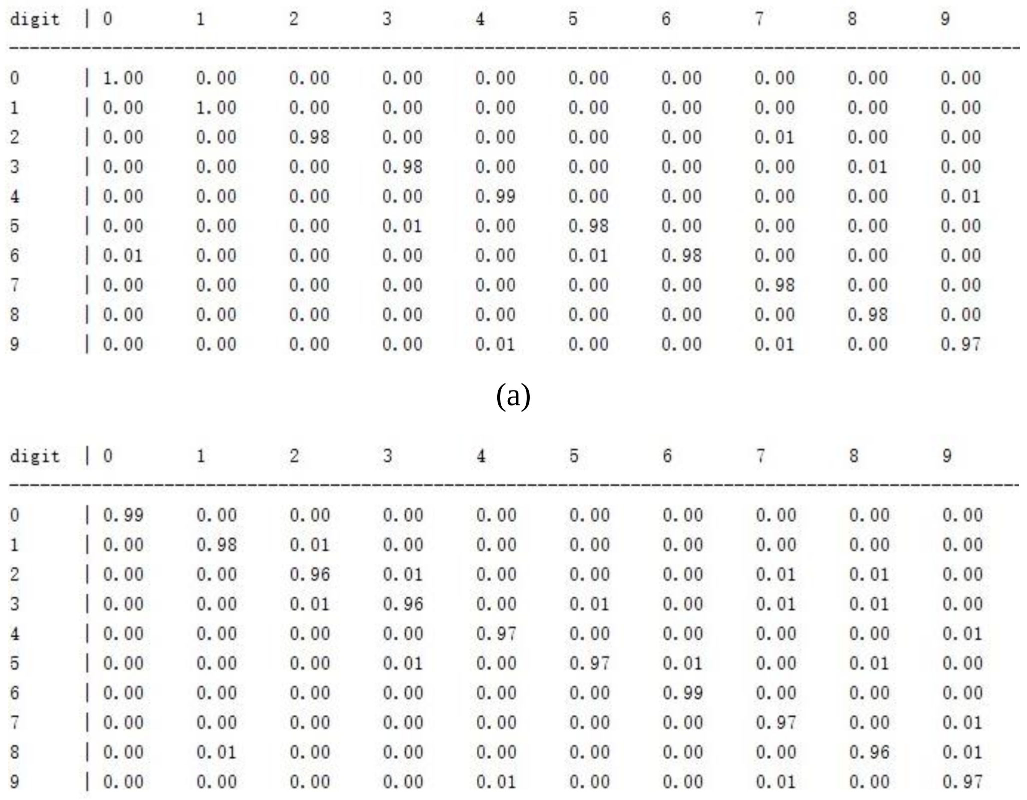


Figure 23: Samples of digit '0' as training data.

Apart from that, to validate the reliability of training database and testing database that obtained from MNIST, two simple tests are carried out. First, the MNIST training database is used to train SVM classifier and test it with MNIST testing database. Second, SVM classifier is trained using MNIST testing database and test it with MNIST training database. The results are computed and tabulated in confusion matrix form as shown in Figure 24 using ‘confusionmat’ function in MATLAB.



```
digit  | 0        1        2        3        4        5        6        7        8        9
------------------------------------------------------------------------------------------------
0      | 1.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
1      | 0.00     1.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
2      | 0.00     0.00     0.98     0.00     0.00     0.00     0.00     0.01     0.00     0.00
3      | 0.00     0.00     0.00     0.98     0.00     0.00     0.00     0.00     0.01     0.00
4      | 0.00     0.00     0.00     0.00     0.99     0.00     0.00     0.00     0.00     0.01
5      | 0.00     0.00     0.00     0.01     0.00     0.98     0.00     0.00     0.00     0.00
6      | 0.01     0.00     0.00     0.00     0.00     0.01     0.98     0.00     0.00     0.00
7      | 0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.98     0.00     0.00
8      | 0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.98     0.00
9      | 0.00     0.00     0.00     0.00     0.01     0.00     0.00     0.01     0.00     0.97
```

(a)

```
digit  | 0        1        2        3        4        5        6        7        8        9
------------------------------------------------------------------------------------------------
0      | 0.99     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
1      | 0.00     0.98     0.01     0.00     0.00     0.00     0.00     0.00     0.00     0.00
2      | 0.00     0.00     0.96     0.01     0.00     0.00     0.00     0.01     0.01     0.00
3      | 0.00     0.00     0.01     0.96     0.00     0.01     0.00     0.01     0.01     0.00
4      | 0.00     0.00     0.00     0.00     0.97     0.00     0.00     0.00     0.00     0.01
5      | 0.00     0.00     0.00     0.01     0.00     0.97     0.01     0.00     0.01     0.00
6      | 0.00     0.00     0.00     0.00     0.00     0.00     0.99     0.00     0.00     0.00
7      | 0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.97     0.00     0.01
8      | 0.00     0.01     0.00     0.00     0.00     0.00     0.00     0.00     0.96     0.01
9      | 0.00     0.00     0.00     0.00     0.01     0.00     0.00     0.01     0.00     0.97
```

(b)

Figure 24: (a) Results of using training database to train SVM (b) Results of using testing database to train SVM classifier

Figure 24 (a) and (b) shows the confusion matrix in percentage form. The columns of the matrix represent the predicted labels, while the rows represent the known labels. As we can see, the recognition accuracy from both tests are very high. Almost all the numbers are correctly identified in both tests. Hence, conclusion can be made that the MNIST database is reliable and suitable to be used as training data.

However, there are some limitations of HOG that will affect the recognition rate. HOG is sensitive to position displacement and size variation, thus all the testing data have to be centered in a 28 x 28 image, with some black pixels cover around the number.

This is to ensure that the testing data is almost similar with the training data in terms of the size and position. However, the average recognition rate of using HOG and SVM alone can only achieve up to 83% on the Score Sheet Template (SST). Hence, salient structure feature was added into classification stage to verify the outcome of SVM.

The ‘predict’ function in MATLAB returns a predicted class labels and a matrix of scores (posterior probabilities) based on trained SVM classifier. The maximum posterior probability means that the sample has the highest similarity score with one of class in training data and this class (0 to 9) is considered as the output of SVM. The scores of posterior probability are ranked from maximum to minimum. Therefore, there are total of 10 rankings as there are 10 classes in training data. For example, SVM classify an input image as digit ‘0’, the image will then by verified by salient structure feature to check whether the image has a big circle. The output will only displayed as ‘0’ when the structural feature is matched with the sample, otherwise it moves to next ranking and continue verifying the structural features till it match until third ranking. The reason of only the top 3 outputs are verified is because about 90% of the handwritten numbers fall in the top three ranking. For the case of all the structural features could not matched with the sample, the sample will be classified as the one with highest similarity score.

4.3 Comparison of Handwritten Recognition Rate of Different Classifiers

The following result is obtained from the testing of two classifiers on images from SST only.

SVM classifier can achieve recognition rate of around 75% to 90%, which is much higher compared with using k-Nearest Neighbours classifier, where the recognition rate is about 65% to 70%. Despite different values of k (5, 10, 30, 36, 50) chosen for kNN classifier, yet the outcome is still unable to achieve accuracy higher than 70%.

Moreover, the execution time for SVM is shorter compared to kNN classifier. However, the recognition rate is improved to around 80% - 95% by verifying the top 3 ranking output from SVM using salient structure feature. The processing time of using SVM and salient structure is almost the same as using SVM alone. The performance of different approaches are tabulated in Table 5 below. In short, the combination of SVM and salient structure has the best performance (highest recognition rate & fastest processing time) compared to other classifier approaches in this project.

Table 5: Recognition rate of handwritten digit using different approaches

Classifiers	Recognition Rate	Processing time
K Nearest Neighbours	65% - 70%	10s – 17s
Support Vector Machine	75% - 90%	8s – 13s
Support Vector Machine + Salient Structure Feature	80% - 98%	8s – 13s

Throughout the testing, several digits were commonly mistaken during the classification stage. Handwritten numbers ‘7’ with a horizontal line in the center as shown in Figure 25 often mistaken as ‘4’ by the classifier.

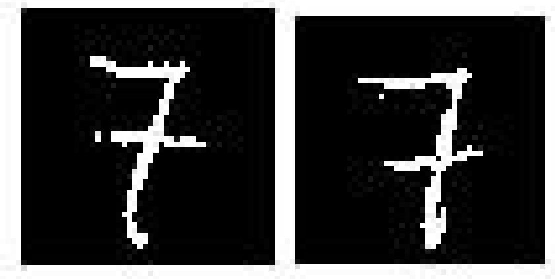
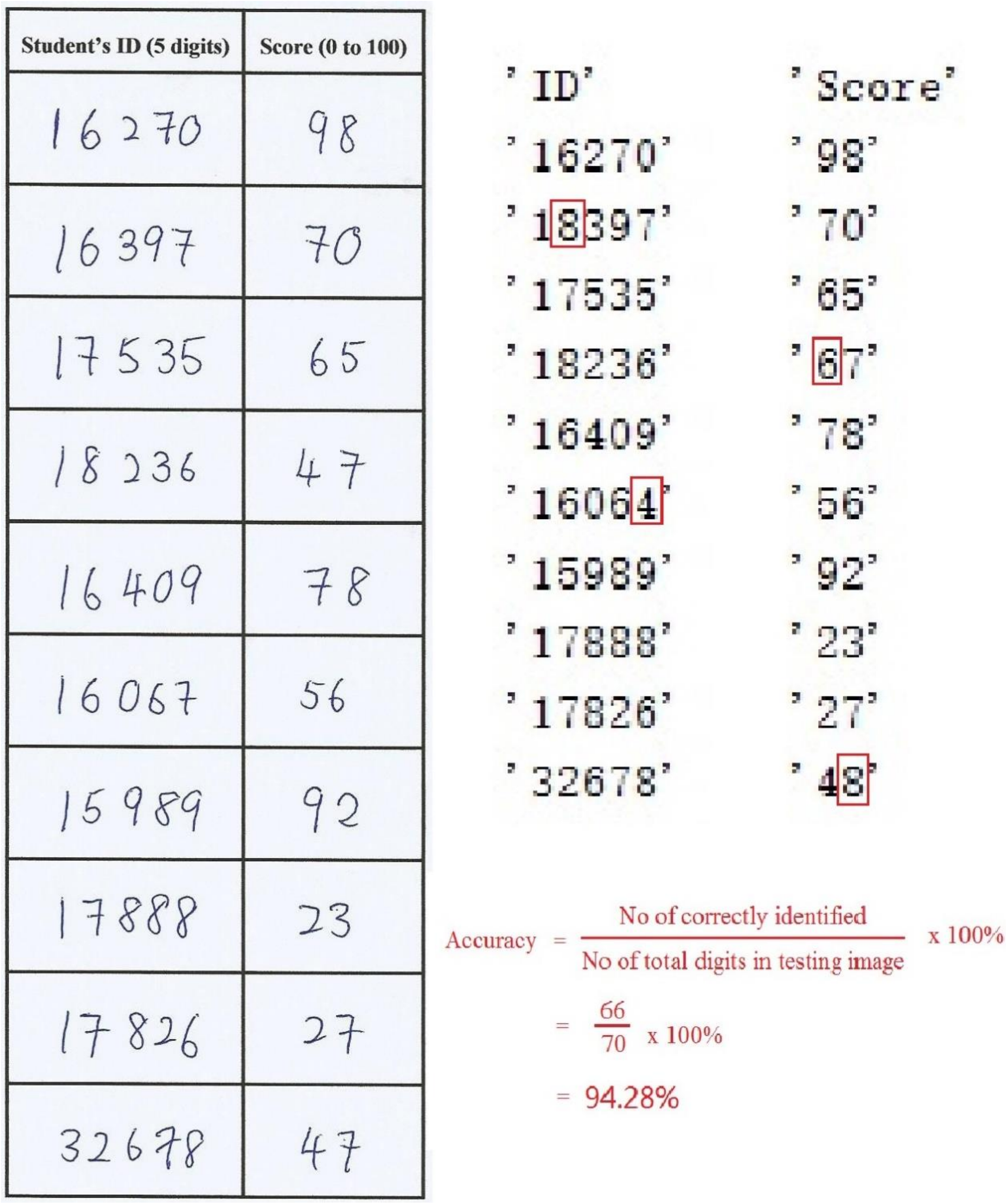


Figure 25: Examples of handwritten '7' that will mistaken as '4'

Besides that, number ‘9’ is often misclassified as ‘7’ and number ‘8’ is often classified as ‘6’. These maybe due to similarities shared between these numbers such as closed loops, vertical line on the right hand side.

Examples of recognition result from SST are shown in Figure 26. The accuracy is observed to be lower when the score sheet is written using pencil instead of pen. The pixels written in pencil does not strongly on paper compared to pen and the pixels have been treated as ‘0’ when the image is converted to binary image.



Student's ID (5 digits)	Score (0 to 100)
16270	98
16397	70
17535	65
18236	47
16409	78
16067	56
15989	92
17888	23
17826	27
32678	47

```
'ID'        'Score'
'16270'     '98'
'18397'     '70'
'17535'     '65'
'18236'     '67'
'16409'     '78'
'16064'     '56'
'15989'     '92'
'17888'     '23'
'17826'     '27'
'32678'     '48'
```

$$\text{Accuracy} = \frac{\text{No of correctly identified}}{\text{No of total digits in testing image}} \times 100\%$$

$$= \frac{66}{70} \times 100\%$$

$$= 94.28\%$$

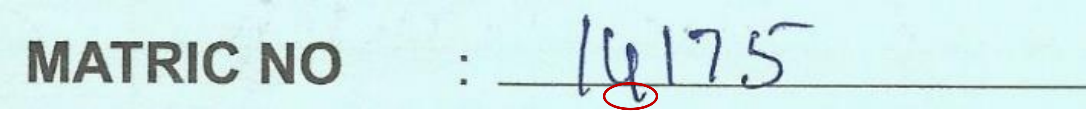
Figure 26: Result of handwritten recognition using SVM classifier and salient structure

4.4 Comparison of Handwritten Recognition Rate on Different Template

As mentioned in the previous section, 2 other templates are introduced on top of SST to test the system accuracy, which are Test Paper and ISDP Final Presentation Assessment. (Refer to Figure 16 and Figure 17)

4.4.1 Test Paper Template

11 samples of ISDP Test Paper are collected for testing purpose and the recognition rate obtained is 70.79%. The recognition rate for this template is the lowest compared to other template used in this project. This is due to some numbers written beyond the area given or part of the number written below the line provided as shown in Figure 27 below.

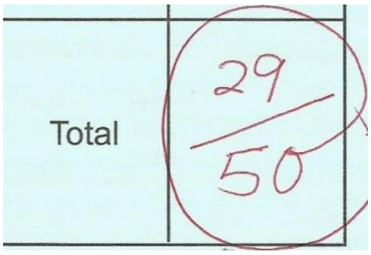


MATRIC NO : 14175

Figure 27: Example of a Matric Number written beyond the area given

For this particular case, the numeral '4' is classified as '0' as the numeral as the vertical writing below the line has been omitted during pre-processing.

There is another limitation for this template as the system is unable to read numbers in fraction as shown in Figure 28. The circle and the horizontal line of the fraction will confuse the program and the horizontal line is even included as an image. More work will need to done to effectively recognize fractions.



Total	29/50

Figure 28: Examples of total score in test paper that cannot be recognized by the system.

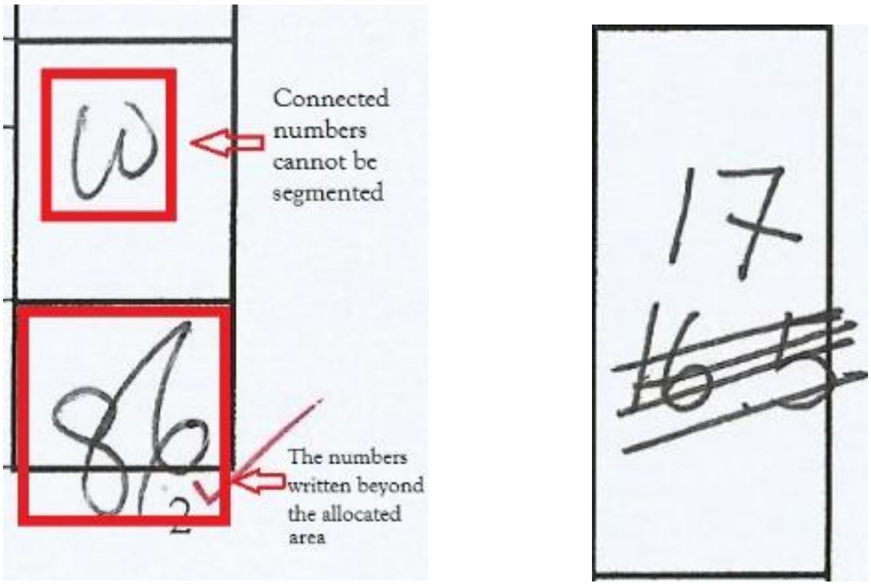
4.4.2 ISDP Final Presentation Assessment

The recognition rate of this template is 78.77%. There are also some limitations for testing this template:

a) Most of the written digits do not fit inside the box allocated as shown in Figure 29 (a) below. This might be due to the area given for lecturer to fill in the marks is small.

b) Another obstacle is connected numbers which cannot be segmented and thus treated as a single image only.

c) Users of this template may also cancel old scores in stroke before adding new scores as shown in Figure 29 (b). The cancelled old score will be identified as an image.


Connected numbers cannot be segmented
The numbers written beyond the allocated area

(a) (b)

Figure 29: (a) Examples of connected numbers and numbers that are written beyond the box given. (b) Old scores cancelled out in stroke.

d) Last but not least, the system also faces difficulties in recognizing decimals as shown in Figure 30. The '.' will be treated as noise and ignored by the system. Hence, in this case the system will classified as '165' instead of '16.5'.

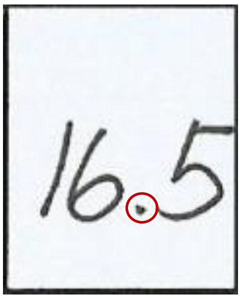


Figure 30: Example of decimals found in the template.

Analysis looking at the processing time from all 3 templates used, the processing time for SST is the longest as the system have to recognize around 140 numbers per Score Sheet. This is in comparison to only around 15 to 20 numbers images that need to be tested either in Test Paper or the ISDP Final Presentation Assessment, which yield shorter processing times.

In a nut shell, the performance of the system is the best for Score Sheet Template (>85%) while the other two templates have yet to achieve more than 85% accuracy due to the limitations pointed out. All performance results are compiled in Table 6 below.

Table 6: Performance of automatic handwritten recognition on different template

Template	**Average Recognition Rate**	**Samples collected**	**Processing Time**
Score Sheet	87%	15	8 - 13s
Test Paper	70.79%	11	1 – 3s
ISDP Final Presentation Assessment	78.77%	27	1 - 4s

However, several improvements can be used to improve accuracy for 2 templates.

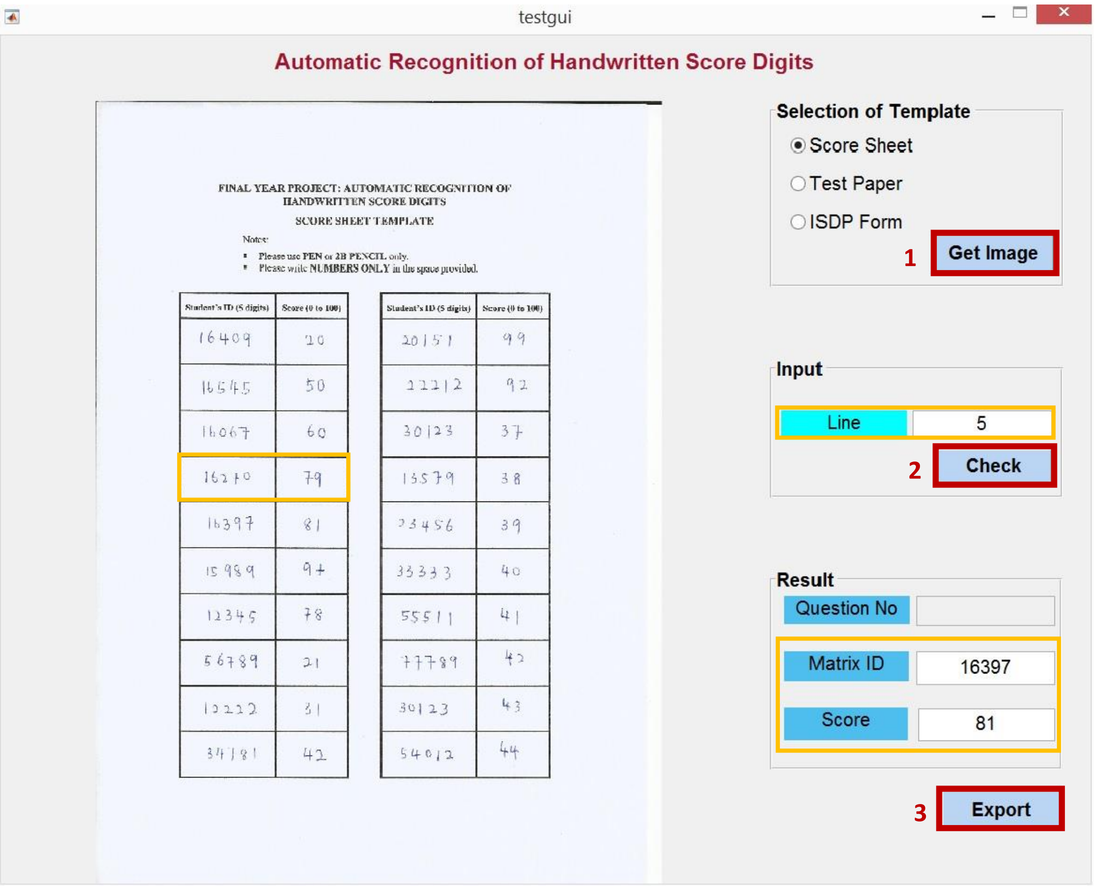
i. Numbers written should remain within the area given.

ii. Check the size of the input images. If one of the image is very large compared to others, omit it. Taking the example from Figure 29 (b) again, the old scores that cancelled out in strokes is treated as a single image as it is connected, thus the area will be much larger compared to digits ‘1’ and ‘7’ within the same area. Thus, the old score can be omitted if the size is unusually large.

iii. To solve the issue for decimals, minimum and maximum values of score can be pre-set before the recognition. For example in Figure 30, maximum value in that column is 20. Since the output is 165 which is bigger than 20, therefore the output can be classified as ‘16.5’ instead of ‘165’.

4.5 Graphical User Interface (GUI)

An interface is developed for this project as shown in Figure 31.



Automatic Recognition of Handwritten Score Digits

FINAL YEAR PROJECT: AUTOMATIC RECOGNITION OF HANDWRITTEN SCORE DIGITS

SCORE SHEET TEMPLATE

Notes:

- Please use PEN or 2B PENCIL only.
- Please write NUMBERS ONLY in the space provided.

Student's ID (5 digits)	Score (0 to 100)	Student's ID (5 digits)	Score (0 to 100)
16409	20	20151	99
16545	50	22212	92
16067	60	30123	37
16270	79	13579	38
16397	81	23456	39
15989	97	33333	40
12345	78	55511	41
56789	21	77789	42
12222	31	30123	43
34181	42	54012	44

Selection of Template

● Score Sheet
○ Test Paper
○ ISDP Form

Input

Line: 5

Result

Question No:
Matrix ID: 16397
Score: 81

Figure 31: GUI for Automatic Recognition of Handwritten Score Digits System

Function of the buttons:

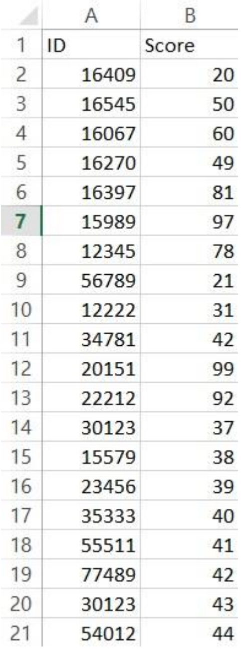
1) **Get Image**: Enable user to select template and then presented with sample images for the template selected.

2) **Check**: Check the handwritten numbers specified by the user and pass the result to 'Result' section.
3) **Export**: Enable user to export the result to a spreadsheet.

First of all, user can select which template to be tested by using the Get Image button.

In the 'Input' section, user can enter which line of numbers to be recognized and the result will be shown in 'Result' section. For example in Figure 31, the input line number is '5' and the system manage to recognize the numbers correctly and show the result in 'Result' section.

Last but not least, the user can export the result and store in Microsoft Excel format as shown in Figure 32.



	A	B
1	ID	Score
2	16409	20
3	16545	50
4	16067	60
5	16270	49
6	16397	81
7	15989	97
8	12345	78
9	56789	21
10	12222	31
11	34781	42
12	20151	99
13	22212	92
14	30123	37
15	15579	38
16	23456	39
17	35333	40
18	55511	41
19	77489	42
20	30123	43
21	54012	44

Figure 32: Examples of scores from SST result stored in Excel file.

CHAPTER 5
CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Handwritten digits recognition is essential in our daily life and yet it is one of the huge challenges in image processing as every person has his/her own handwriting style. An enormous work has been done in handwritten digit recognition, but have yet to achieve a module with no mistake at all. In this project, a handwritten digit recognition system is proposed using HOG as the feature extraction method.

Pre-processing step is very important as it will enhance the performance of feature extraction later. However, connected numbers are still cannot be segmented properly. In addition, some important parts of the number that we want to recognize will be removed if the threshold value of pixels is not suitably chosen.

In addition, training data is one of the crucial part in this project as the recognition rate is highly dependent on the training data. MNIST testing database which contains about 10 000 samples is used to train the classifier. Salient structure feature is used after SVM in the classification stage to increase the recognition rate of handwritten digits.

In brief, an automatic handwritten digits recognition system is developed using HOG as feature extraction method and SVM as classifier. However, salient structural feature is added in classification stage to improve the recognition rate. For SST, the average processing time is 10 seconds and average recognition rate is 87%. However for the other templates, the processing time is much lower but yields lower accuracy and this can be further improved.

5.2 Recommendations

Several recommendations are suggested:

i. Classification Method and Pre-Processing

   Future research can be conducted to find a better classification method such as Neural Networks for the features extracted by HOG. Pre-processing also can be improved further to be able to read numbers written beyond the area given.

ii. GUI

   The design of GUI can be further improved to become more user-friendly and more templates are available for user to select.

iii. Decimals / Fractions

   Maximum values of score in a certain area can be pre-determined to solve the difficulties in recognizing decimals. Another solution is by adding extra columns for decimals such as application in cheque or invoice. Similar solution can be used when dealing with numbers in fraction as well, special columns can be designed to let the user to fill in the numerator and denominator.

REFERENCES

[1] R. Plamondon and S. N. Srihari, “Online and off-line handwriting recognition: a comprehensive survey,” *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 22, no. 1, pp. 63–84, 2000.

[2] L. Eikvil, “Optical character recognition,” *citeseer. ist. psu. edu/142042. html*, 1993.

[3] S. Mori, H. Nishida, and H. Yamada, *Optical Character Recognition*. New York, NY, USA: John Wiley &amp; Sons, Inc., 1999.

[4] A. Sinha, “An improved recognition module for the identification of handwritten digits,” Citeseer, 2002.

[5] T. Ahonen, A. Hadid, and M. Pietikainen, “Face Description with Local Binary Patterns: Application to Face Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[6] L. L. Ghai, “Automatic Assessment Mark Entry System Using Local Binary Pattern (LBP),” Universiti Teknologi Petronas, 2014.

[7] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” *2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, pp. 886–893, 2005.

[8] S. Bhowmik, M. G. Roushan, R. Sarkar, M. Nasipuri, S. Polley, and S. Malakar, “Handwritten Bangla Word Recognition Using HOG Descriptor,” *2014 Fourth Int. Conf. Emerg. Appl. Inf. Technol.*, pp. 193–197, 2014.

[9] W. Y. Xiang and P. Sebastian, “Handwriting recognition using webcam for data entry,” in *Signal Processing & Its Applications (CSPA), 2015 IEEE 11th International Colloquium on*, 2015, pp. 100–105.

[10] O. Chapelle, “Support vector machines for histogram-based image classification,” *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1055–64, 1999.

[11] L. E. Peterson, “K-nearest neighbor,” *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[12] S. Noh, “χ 2 Metric learning for nearest neighbor classification and its analysis,” in *Pattern Recognition (ICPR), 2012 21st International Conference on*, 2012, pp. 991–995.

[13] M. Cheriet, N. Kharma, C. Liu, and C. Suen, *Character Recognition Systems: A Guide for Students and Practitioners*. Wiley-Interscience, 2007.

[14] Y. LeCun, C. Cortes, and C. J. C. Burges, “The MNIST database of handwritten digits, 1998,” *Available Electron. http//yann. lecun. com/exdb/mnist*, 2012.

APPENDIX I

FINAL YEAR PROJECT: AUTOMATIC RECOGNITION OF HANDWRITTEN SCORE DIGITS

SCORE SHEET TEMPLATE

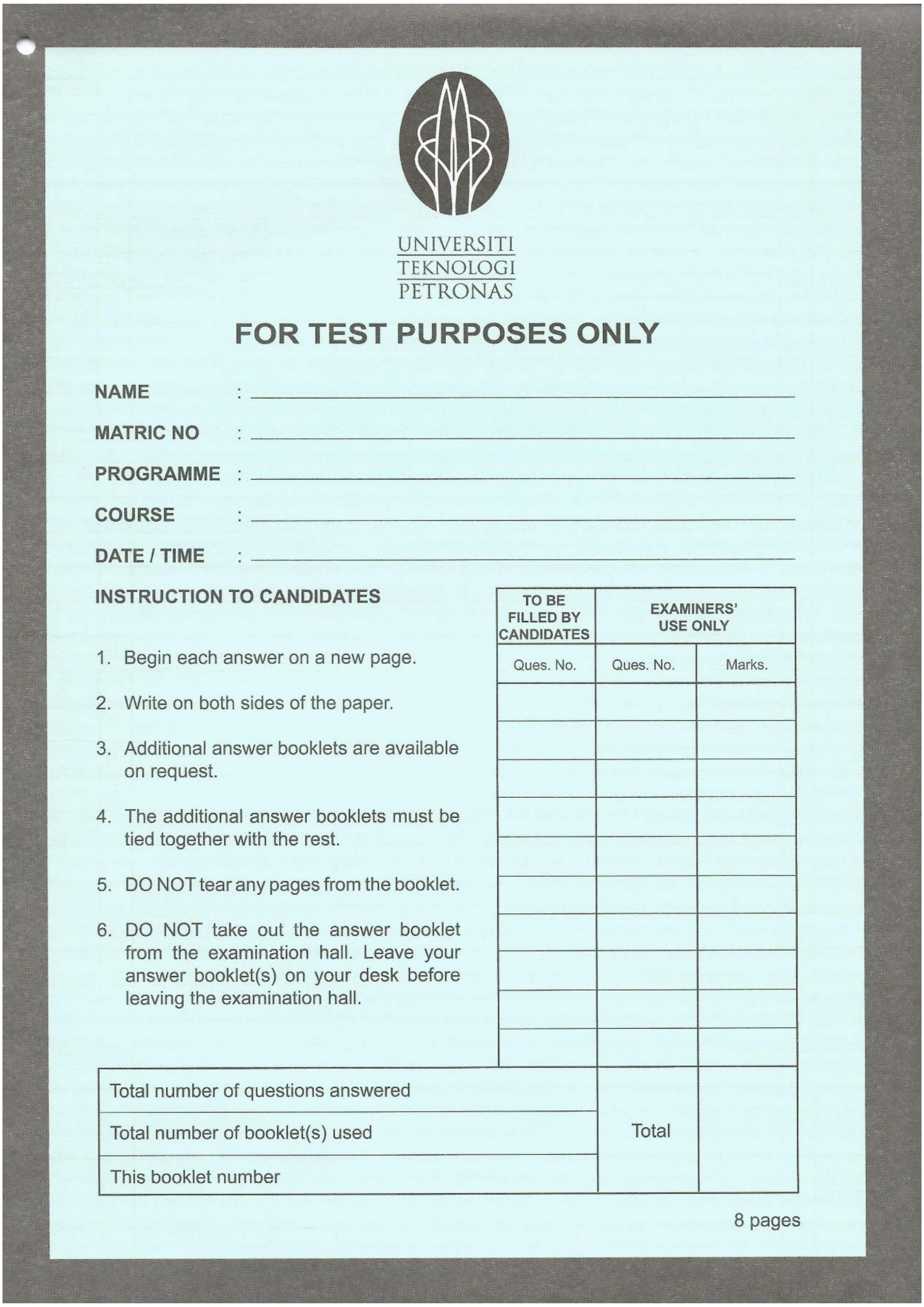
Notes:

- Please use **PEN** or **2B PENCIL** only.
- Please write **NUMBERS ONLY** in the space provided.

Student’s ID (5 digits)	Score (0 to 100)

Student’s ID (5 digits)	Score (0 to 100)

APPENDIX II



UNIVERSITI
TEKNOLOGI
PETRONAS

FOR TEST PURPOSES ONLY

NAME : ____________________

MATRIC NO : ____________________

PROGRAMME : ____________________

COURSE : ____________________

DATE / TIME : ____________________

INSTRUCTION TO CANDIDATES

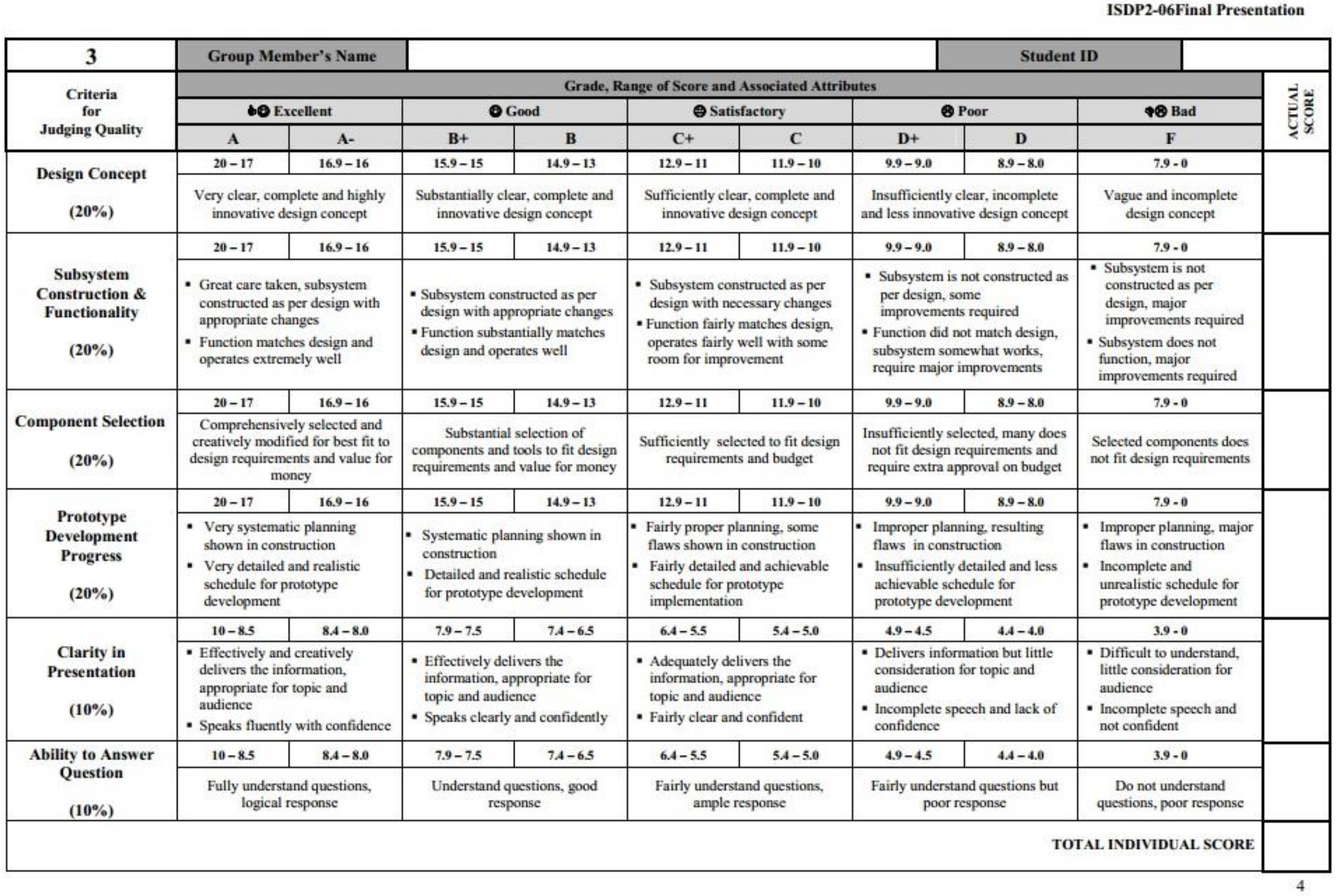
1. Begin each answer on a new page.
2. Write on both sides of the paper.
3. Additional answer booklets are available on request.
4. The additional answer booklets must be tied together with the rest.
5. DO NOT tear any pages from the booklet.
6. DO NOT take out the answer booklet from the examination hall. Leave your answer booklet(s) on your desk before leaving the examination hall.

TO BE FILLED BY CANDIDATES	EXAMINERS' USE ONLY	
Ques. No.	Ques. No.	Marks.

Total number of questions answered	Total	
Total number of booklet(s) used		
This booklet number		

8 pages

APPENDIX III



ISDP2-06Final Presentation

3	Group Member's Name						Student ID			
Criteria for Judging Quality	Grade, Range of Score and Associated Attributes									ACTUAL SCORE
	👍☺ Excellent		☺ Good		😐 Satisfactory		☹ Poor		👎☹ Bad	
	A	A-	B+	B	C+	C	D+	D	F	
Design Concept (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	Very clear, complete and highly innovative design concept		Substantially clear, complete and innovative design concept		Sufficiently clear, complete and innovative design concept		Insufficiently clear, incomplete and less innovative design concept		Vague and incomplete design concept	
Subsystem Construction & Functionality (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	▪ Great care taken, subsystem constructed as per design with appropriate changes ▪ Function matches design and operates extremely well		▪ Subsystem constructed as per design with appropriate changes ▪ Function substantially matches design and operates well		▪ Subsystem constructed as per design with necessary changes ▪ Function fairly matches design, operates fairly well with some room for improvement		▪ Subsystem is not constructed as per design, some improvements required ▪ Function did not match design, subsystem somewhat works, require major improvements		▪ Subsystem is not constructed as per design, major improvements required ▪ Subsystem does not function, major improvements required	
Component Selection (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	Comprehensively selected and creatively modified for best fit to design requirements and value for money		Substantial selection of components and tools to fit design requirements and value for money		Sufficiently selected to fit design requirements and budget		Insufficiently selected, many does not fit design requirements and require extra approval on budget		Selected components does not fit design requirements	
Prototype Development Progress (20%)	20 – 17	16.9 – 16	15.9 – 15	14.9 – 13	12.9 – 11	11.9 – 10	9.9 – 9.0	8.9 – 8.0	7.9 - 0	
	▪ Very systematic planning shown in construction ▪ Very detailed and realistic schedule for prototype development		▪ Systematic planning shown in construction ▪ Detailed and realistic schedule for prototype development		▪ Fairly proper planning, some flaws shown in construction ▪ Fairly detailed and achievable schedule for prototype implementation		▪ Improper planning, resulting flaws in construction ▪ Insufficiently detailed and less achievable schedule for prototype development		▪ Improper planning, major flaws in construction ▪ Incomplete and unrealistic schedule for prototype development	
Clarity in Presentation (10%)	10 – 8.5	8.4 – 8.0	7.9 – 7.5	7.4 – 6.5	6.4 – 5.5	5.4 – 5.0	4.9 – 4.5	4.4 – 4.0	3.9 - 0	
	▪ Effectively and creatively delivers the information, appropriate for topic and audience ▪ Speaks fluently with confidence		▪ Effectively delivers the information, appropriate for topic and audience ▪ Speaks clearly and confidently		▪ Adequately delivers the information, appropriate for topic and audience ▪ Fairly clear and confident		▪ Delivers information but little consideration for topic and audience ▪ Incomplete speech and lack of confidence		▪ Difficult to understand, little consideration for audience ▪ Incomplete speech and not confident	
Ability to Answer Question (10%)	10 – 8.5	8.4 – 8.0	7.9 – 7.5	7.4 – 6.5	6.4 – 5.5	5.4 – 5.0	4.9 – 4.5	4.4 – 4.0	3.9 - 0	
	Fully understand questions, logical response		Understand questions, good response		Fairly understand questions, ample response		Fairly understand questions but poor response		Do not understand questions, poor response	
									TOTAL INDIVIDUAL SCORE	

4