FINGER VEIN RECOGNITION BASED ON AN IMPROVED K-NEAREST CENTROID NEIGHBOR CLASSIFIER

NG YEE WEI

UNIVERSITI SAINS MALAYSIA

2017

FINGER VEIN RECOGNITION BASED ON AN IMPROVED K-NEAREST CENTROID NEIGHBOR CLASSIFIER

by

NG YEE WEI

Thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering (Electronic Engineering)

JUNE 2017

ACKNOWLEDGEMENT

Upon the completion of this project, I had received a lot of assistance and support in various aspects from people surrounding me. Therefore, I would like to take this opportunity to express my deepest appreciation to those people.

First and foremost, I would like to show greatest gratitude to my final year project supervisor, Associate Professor Dr Bakhtiar Affendi Rosdi, who is very responsible and had provided clear guidance to me throughout the implementation of this project. He played his role as supervisor very well by giving me direction to start the project and useful suggestion to solve problems encountered as well as providing feedback to improve the project and thesis. With his aid, I was able to complete the project within period given.

Next, I wish to thank to my colleagues who were willing to share beneficial knowledge with me to develop the project. They showed no hesitation to help me whenever I faced some technical problems in doing this project. Their sincere comment to the project has resulted in improvement of this project.

Lastly, special thanks to my family who always give me mentally support during the project implementation. Their continuous support and believe in me had led my persistence to complete this project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ATM	Automated Teller Machine
CLAHE	Contrast limited adaptive histogram equalization
KECA	Kernel Entropy Component Analysis
KNCN	K-Nearest Centroid Neighbor
KNN	K-Nearest Neighbor
KPCA	Kernel Principal Component Analysis
LMKNCN	Local Mean K-Nearest Centroid Neighbor
LMKNN	Local Mean K-Nearest Neighbor
MRSKNCN	Modified Repeatedly Selected K-Nearest Centroid Neighbor
NCN	Nearest Centroid Neighbor
NIR	Near infrared
NN	Nearest Neighbor
PCA	Principal Component Analysis
PIN	Personal Identification Number
ROI	Region of Interest
RSKNCN	Repeatedly Selected K-Nearest Centroid Neighbor
SRC	Sparse Representation Classifier
SVM	Support Vector Machine

PENGECAMAN URAT JARI BERDASARKAN PENGELAS JIRAN SENTROID K TERDEKAT YANG TELAH DITAMBAH BAIK

ABSTRAK

Projek ini dijalankan untuk mengusulkan pengelas jiran sentroid K terdekat (KNCN) yang telah ditambah baik untuk pengecaman urat jari. Kebelakangan ini, pengecaman urat jari menjadi salah satu teknologi biometrik yang terkenal untuk diguna dalam pelbagai applikasi disebabkan ciri-ciri urat jari. Beberapa pengelas telah dicadangkan untuk proses klasifikasi dalam sistem tersebut. Berbanding dengan pengelas lain, KNCN mempunyai kekuatannya kerana mempertimbangkan jarak dan pengagihan ruang. Namun, kekuatan ini menjadi kelemahannya kerana pengelas berkemungkinan terlebih menganggar julat NCN yang bakal dipilih. Selain itu, pemberat bagi setiap jiran sentroid terdekat tidak dipertimbangkan oleh pengelas KNCN dalam proses mengundi dan masa pemprosesan juga meningkat apabila nilai *k* yang besar dipilih. Oleh itu, pengelas KNCN yang lebih baik dan mempertimbangkan semua masalah yang dibincang di atas telah diusulkan untuk pengecaman urat jari dalam projek ini. Ianya dijalankan dengan menganalisa dan mengubahsuai pengelas KNCN asal supaya pengelas tersebut ditambahbaik dari segi ketepatan dan masa pemprosesan. Berdasarkan cara permilihan NCN yang baru , pengelas RSKNCN telah diusulkan dan mencapai 87.64% ketepatan (4.34% lebih tinggi daripada pengelas KNCN asal) atas pangkalan data FV-USM. Versi RSKNCN yang diubahsuai menunjukkan 87.06% ketepatan dengan prestasi masa 182.94 milisaat/sampel. Walaupun ketepatannya dikurangkan sebanyak 0.58% berbanding dengan RSKNCN asal, tetapi masa pemprosesannya hanya 0.3 kali ganda daripada RSKNCN asal. Secara keseluruhanmya, projek ini berjaya menghasilkan pengelas KNCN yang telah ditambahbaik dan mencapai keseimbangan antara ketepatan dan prestasi masa untuk pengecaman urat jari.

FINGER VEIN RECOGNITION BASED ON AN IMPROVED K-NEAREST CENTROID NEIGHBOR CLASSIFIER

ABSTRACT

This project is developed to propose an improved K-Nearest Centroid Neighbor classifier for finger vein recognition. Recently, finger vein recognition has become one of the most popular biometric technologies to be used in various applications due to finger vein's properties. Several classifiers have been proposed for the classification process in finger vein recognition system. Compared to other classifiers, KNCN has advantage of considering both proximity and spatial distribution. However, this becomes a disadvantage as it may overestimate the range of NCN to be chosen. In addition, in a typical KNCN classifier, the weightage of each nearest centroid neighbor is not considered in the voting process. Besides, the classifier processing time increases when a large value of k is chosen. Therefore, an improved KNCN classifier that considers those problems is proposed for finger vein recognition in this project. This is done by analyzing the typical KNCN classifier and applying modification on it to improve its performance in term of accuracy and processing time. Based on a new NCN selection method proposed, RSKNCN classifier had been proposed and had achieved finger vein recognition rate of 87.64 % on FV-USM database which is 4.34 % higher than the accuracy of a typical KNCN classifier. Modified version of RSKNCN classifier had improved the processing time performance by achieving accuracy of 87.06 % with 182.94 ms/sample processing time performance. Although there is 0.58 % drop in accuracy compared to RSKNCN classifier, the processing time performance had shortened to 0.30 times of the processing time of RSKNCN classifier. Overall, this project has successfully developed an improved KNCN classifier which achieved balance performance between accuracy and processing time in finger vein recognition.

CHAPTER 1

INTRODUCTION

1.1 Background

In this modern era, biometric technology [1] that is based on individuals' unique physiological and behavior features is widely used in various applications. Mechanical lock system, smart card based user authentication system and PIN number based ATM transaction will be a thing of pass due to convenience and high level of security of this technology. In those applications, biometric technology is either applied as verification or identification system [2]. In verification process, query sample is compared with claimed identity's templates that are stored in database whereas identification is process of finding a match for query sample by compare it with every training samples in database.

Iris [3], face [4], fingerprint [5], handwriting [6], gait [7] and ear recognition [8] are some of techniques that have been applied in biometric recognition. However, there are some weakness exist in those techniques that make the recognition system less reliable. For example, handwriting and gaits can be easily counterfeit, fingerprint is prone to damage, facial recognition is easily affected by lighting condition [9], and faces' features are not permanent and sometimes not unique as well.

Apart from the techniques stated above, there is another popular biometric technique that is introduced to public which is finger vein recognition [10]. This technique has overcome the above- mentioned problems that face by other techniques. Due to properties of finger vein such as unique, permanent and anti- counterfeit that make it reliable to be used in human recognition system [9], finger vein recognition has receiving a lot of attention among researchers, Other than that, finger vein image acquisition process is more preferred by public compared to the commonly used fingerprint recognition because it solves the problem of those who had unclear fingerprint and it is also more hygienic due to its contactless process.

As mentioned in **[**11**]**, finger vein identification model is implemented starting from capturing images using near infrared (NIR) camera and image processing on the taken images. After that, the images are undergoing feature extraction to extract unique features of each image, and finally classification process takes over. Classifier is one of the crucial factors that decide performance of the identification system.

Therefore, this project is to improve one of the existed classifiers which is K-Nearest Centroid Neighbor (KNCN) classifier, so that it performs better in finger vein recognition. As identity forging technology is getting more advanced nowadays, its anti-forging technology should follow its step too in order to prevent criminal from happen and protect the security of the world that lived by human being. With a better recognition rate and more reliable finger vein recognition system introduced, people will no longer have anxiety about their own safety as well as the security of their precious property. In term of convenience, people no longer need to worry about problem of forgetting password or PIN number and other difficulties, such as unclear fingerprint and hygienic problem, to prove their identity for user authentication system with a better biometric technology.

1.2 Problem Statement

As finger vein recognition gains popularity in biometric technology, various works had been done by researchers to achieve higher identification accuracy. Classification is a vital procedure to recognize identity of a person through his or her finger vein. Several classifiers such as Naïve Bayes [12], Support Vector Machine (SVM) [13], Sparse Representation Classifier (SRC) [14], K-Nearest Neighbor (KNN) [12] [15] [16], and K-Nearest Centroid Neighbor (KNCN) [15] [16] have been applied in finger vein recognition.

However, each classifier has its pros and cons. Sparse representation based classifier (SRC) that proposed by Chen and Wang in [13] and SVM proposed by Bai and Prabi in [14] have achieved high accuracy in finger vein recognition, but they are complex and time consuming. Based on comparison of classifiers made by [15] and [16] on finger vein recognition, KNN has lower recognition rate compared to KNCN and other classifiers, but due to its simplicity, it is still preferred by many researcher in their work.

Among those classifiers, KNCN has potential to become the most suitable classifier for finger vein identification system. KNCN, which is an extension from KNN, has better accuracy performance than KNN using Euclidean distance [17], takes less processing time than SVM and SRC [17]. However, typical KNCN classifier neglects weightage of each nearest centroid neighbor (NCN) in voting process [18] [19] [20]. In some cases, it may also overestimate the range of nearest NCN to be chosen [19] [20]. Besides, its processing time also increases when large value of k is chosen. These problems have lowered its classification performance.

Previously, few studies have been made by focusing on improving its accuracy performance. For example, LMKNCN [21] [20] and WKNCN [18] are the successful improvement from KNCN classifier by apply additional algorithm on it. This has proven that accuracy performance of KNCN has high potential to be further improved. However, both LMKNCN and WKNCN do not improve processing time of the classifier.

Therefore, an improved KNCN classifier is proposed in order to enhance accuracy of finger vein recognition to a higher level, at the same time, reduce the processing time of the system.

1.3 Objectives

The objectives of this project are:

i. To improve accuracy performance of finger vein recognition system by improving K-Nearest Centroid Neighbor classifier.

ii. To reduce processing time of the improved K-nearest centroid neighbor classifier without affecting its accuracy performance on finger vein recognition

1.4 Project Scope

This project discusses about development of finger vein recognition based on an improved KNCN classifier. The main concern of this project is on classification process in identification system.

This project involves analysis of typical KNCN classifier about its characteristics and relationship between important factors in the classifier. Based on the analysis, new NCN selection method and voting method is proposed by setting hypothetical rules and conditions to those related factors to improve performance of typical KNCN classifier. Performance of the improved KNCN classifier is compared with previously proposed classifiers on finger vein recognition system and is justified based on two parameters which are accuracy and processing time.

Besides, feature extraction on finger vein images is also included in this project. Before classification process, unique feature of each finger vein image is extracted by applying PCA to reduce image dimension and discard common data among those images. However, finger vein image acquisition, image processing (ROI extraction) and image enhancement is skipped in this project as FV-USM [22] which is an existing database is used.

1.5 Thesis Outline

This thesis outlines finger vein recognition based on an improved K-Nearest Centroid Neighbor. There are a total of five chapters which are introduction, literature review, methodology, results and discussions, and conclusion.

Chapter 2 of this thesis presents literature review about previous works that is related to this project. Biometric technology, general model of finger vein recognition system, principal component analysis (PCA) and some classifiers that have been used in finger vein recognition system are analyzed in this chapter.

Chapter 3 explains the methodology of this project. Procedures and flow charts that are involved in this project starting from the analysis of a typical KNCN classifier, improvement of the classifier and classifier's performance justification are clearly described in this chapter.

Chapter 4 of the thesis is about the results and discussion of this project. Data collected and analyses made are displayed in table and graphical presentation. Comparisons and discussions of the obtained results are also discussed in this chapter.

Lastly, chapter 5 concludes the whole project by indicating whether the project's objectives are achieved based on the overall results of project. Project limitation and future work are also included in this chapter.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

Nowadays, biometric technology is widely practiced in the applications that we used in daily life and it slowly replaces the traditional authentication system. Among the biometric techniques, finger vein recognition has attracted interest of researchers from all over the world due to its special properties. The advantages of recognition system using finger vein compared with other biometric traits is clearly shown in section 2.2 of this chapter.

General model of finger vein recognition system is studied as a benchmark for project implementation. Each part of the model will be discussed in detail in section 2.3. Next, part of the system model which is feature extraction using principal component analysis is further explained in section 2.4.

In section 2.5, previous works that related to finger vein image classification are reviewed. Various types of classifiers that had been used for finger vein identification is introduced. Advantage and disadvantage of each classifier is also described in this section.

Next, section 2.6 discusses the KNCN classifier which is the main focus of this project. Basic concept of KNCN classifier and problems existed in it are explained in this section. Lastly, literature review for this project is summarized in section 2.7.

2.2 Advantages of Finger Vein Recognition Technology Compared with Other Biometric Traits

Over the past decade, various biometric techniques have been continuously proposed by researchers. Iris [3], face [4] and fingerprint [5] are some examples of biometric recognition that had been introduced. Among those techniques, finger vein recognition is more reliable and practical compared to other recognition system to be widely applied in user authentication, access control, forensics and financial transaction [9].

First, finger vein patterns are unique. Each finger of individuals has different vein pattern that lies beneath skin that is invisible and permanent [9]. Unlike fingerprint, it is difficult to forge and its characteristic will not be affected by the damage on outer skin. In [23], it is mentioned that about 5% of fingerprint can hardly be collected due to physiological defect. Besides, finger vein patterns are only available in living body, and it is impossible to steal identity of a dead person [9].

Although finger vein is hidden under skin, but its pattern is easy to be captured using infrared light. Even with the use of low resolution camera, stable and clear vein pattern can be obtained [24]. Compared with iris recognition where the image acquisition is difficult due to poor image quality and the position of iris [2], finger vein recognition is perhaps a better choice although both of them are unique and permanent [2]. In addition, finger vein recognition is more preferable than hand dorsal vein due to its small size that makes image acquisition become easier [25].

Furthermore, finger vein patterns acquisition process is contactless as the device use infrared light to capture vein images [24]. This is to ensure hygiene of the image obtaining process so that it is convenient and clean for users.

2.3 General Model of Finger Vein Recognition System

A general model of finger vein recognition system is as shown in Figure 2.1. The model starts with training sample images enrolment and input image (testing sample) acquisition. In this stage, finger vein pattern is captured by NIR (near infrared) camera in grayscale. NIR [11] is a type of spectroscopy technique. Infrared thermograph is a non-destructive technique delivering temperature images of body.

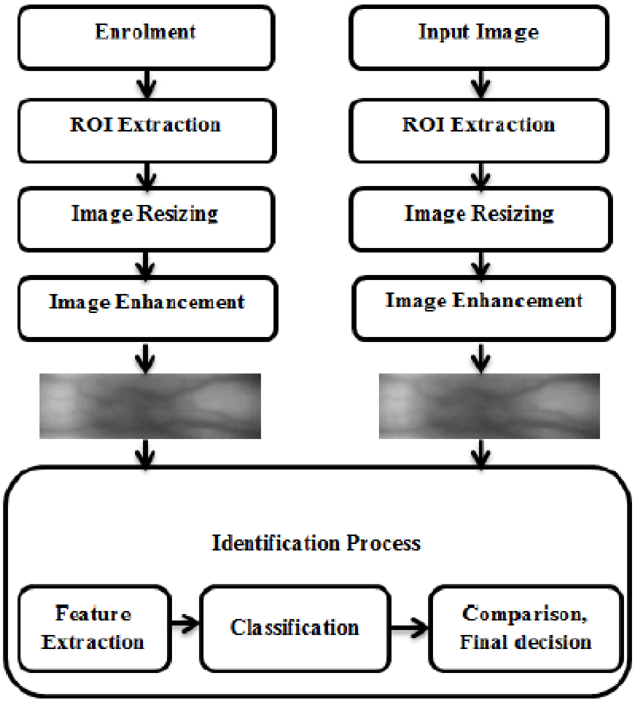
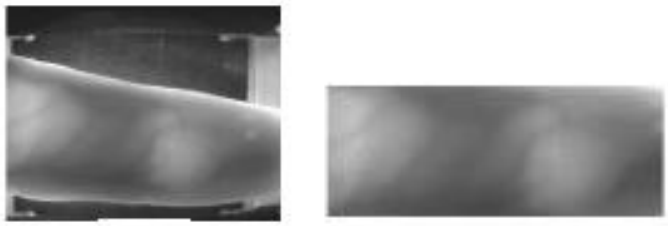


Figure 2.1: Architecture of finger vein recognition system [18]

In the next stage, the images obtained undergoes image processing process which consists of ROI extraction [26] and image resizing. For ROI extraction, it removes the unwanted black background that will affect the accuracy of the system. The cropped images are resized to an appropriate pixel per image to reduce execution time and minimize noise with minimum effect to the accuracy of system. Example of finger vein image before and after ROI extraction is shown in Figure 2.2.



(a) (b)

Figure 2.2: (a) Before ROI extraction (b) After ROI extraction [25]

As stated in [18], the captured image usually has low contrast. Therefore, the image requires to go through image enhancement, for example, by using modified Gaussian high pass filter and Contrast limited adaptive histogram equalization (CLAHE) [27], before storing in database. Finger vein image before and after image enhancement by modified Gaussian high pass filter is shown in Figure 2.3.

(a) (b)

Figure 2.3: (a) Before image enhancement (b) After image enhancement [18]

Next, the image in database is sent to identification process which consists of feature extraction, classification, and final decision. The purpose of feature extraction is to remove common features between all samples and extract unique features of each sample to improve accuracy of result. Examples of techniques [28] [29] used are PCA, KPCA and KECA. After that, the images are classified using any type of classifiers which will be further discussed in section 2.5.

Lastly, comparison and final decision is made. Performance of classifier is measured based on two parameters which are accuracy (in percentage) and processing time (in millisecond).

2.4 Principal Component Analysis

Principal component analysis is one of techniques of feature extraction that frequently been used in biometric technology especially in face recognition system **[**30**]**. In **[**10**]** and **[**29**]**, PCA is also used for feature extraction in finger vein recognition system. PCA is a tool for analyzing data, extracting important information from a set of multivariate training data, and transforming the data into a new coordinate system **[**28**]**. For high dimensional image, PCA can transfer it into a lower dimensional image which only consist important information. Steps to generate PCA are explained as follow **[**31**]**:

Step 1: Load a training set. Consider each image has N x N pixels, convert pixel data of each image into N^2 x 1 as column of a vector.

Step 2: Calculate average value of the training set data and subtract the average value form each element in the vector found in step 1.

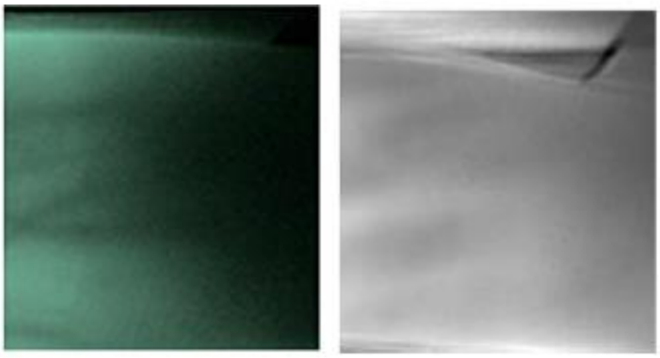
Step 3: Calculate the covariance matrix of the difference value found in step 2.

Step 4: Calculate eigenvectors and eigenvalues from the covariance matrix in step 3

Step5: Select *K* best eigenvalues that represent the whole training data. First principle component represent the eigenvector with highest eigenvalue and so on for the second.

Step 6: Convert original data to a lower dimensional of K eigenvectors.

Improvements from PCA which are KPCA and KECA, are introduced in **[**28**]** and **[**27**]** for feature extraction in finger vein recognition system to obtain better accuracy of recognition. However, PCA is still preferred in many of recognition system. Thus, in this project, PCA also be used in feature extraction due to its simplicity. Finger vein image before and after feature extraction by PCA is compared in Figure 2.4.



(a) (b)

Figure 2.4: (a) Original finger vein image (b) Finger vein image after PCA (four feature vector) [29]

2.5 Related Works on Classifiers for Finger Vein Recognition

This section elaborates on classifiers that proposed previously for finger vein recognition system. The classifiers to be discussed are Naïve Bayes [12], SVM [13], SRC [14] and KNN [12] [16] [15].

2.5.1 Naïve Bayes

In [12], Naïve Bayes is used for classification in the model of finger vein recognition system. It is a classifier that works based on Bayes rule of probability theory which predicts a given data's class by matching given data to class with highest posterior probability. Naïve Bayes conditional independence assumption considers all attributes in a training samples are independence of each other. Classification using Naïve Bayes classifier is described as follow [32]:

Consider a set of training dataset, D={$X^{(1)}$, …, $X^{(n)}$} that consists of n instances, where each instance, $\mathrm{X} = \{x_1, \dots, x_n\} \in D$ is represented in n-dimensional vector and are labeled with class $c_j = \{c_1, \dots, c_m\} \in C$. Probability of a training instance belonging to a class, $\mathrm{P}(c_j|\mathrm{X})$ is calculated using Eq. (2.1). $P(c_j)$ is class prior probability and $P(X)$ is predictor prior probability.

$$\mathrm{P}(c_j|\mathrm{X}) = \frac{\mathrm{P}(\mathrm{X}|c_j)\, P(c_j)}{P(X)} \tag{2.1}$$

However, work to compute $\mathrm{P}(\mathrm{X}|c_i)$ has high difficulty due to insufficient data. Therefore, Naïve Bayes independence assumption as Eq. (2.2) is made.

$$\mathrm{P}(\mathrm{X}|c_j) = \prod_{i=1}^{n} P(x_i|c_j) \tag{2.2}$$

where x_i is the ith attribute value of instance X, i =1, 2, …., n

In training stage, $P(x_i|c_j)$ and $P(c_j)$ for each class c_j and each attribute value x_i is estimated for a given testing instance, $T = \{t_1, \dots t_m\}$ where t_m is an attribute value of the testing instance. In classification stage, Naïve Bayes classifier classifies the testing instance using Eq. (2.3). Class of instance in D, C_{NB} with highest $P(x_i|c_j)P(c_j)$ value will be assigned to the testing instance.

$$C_{NB}(t) = \operatorname{argmax}_{c_j \in C} \prod_{i=1}^{n} P(x_i|c_j)\, P(c_j) \tag{2.3}$$

Naïve Bayes classifier is well known for its simplicity and performance in noisy environment [32]. However, due to the conditional independence assumption, it seems less practical to deal with real-world data because it does not consider correlations and dependencies in features. Research in [33] had showed that vein recognition using Naïve Bayes has lower accuracy (80%) then other classifiers such as SVM and KNN (above 90%) . To minimize this disadvantage, weighting method is introduced on Naïve Bayes classifier to improve its classification performance [32].

2.5.2 Support Vector Machine

Support Vector machine (SVM) is one of the popular choice for classification in finger vein pattern recognition system **[**25**]** **[**13**]**. SVM classifier constructs a set of hyper-plane in multidimensional space based on data provided and classify based on that surface which separates positive training samples from negative ones with larger margin **[**34**]**. The optimal hyper-plane is determined to maximize the generalization ability of classifier and it can be found by applying optimization theory. Classification method using SVM for linear separation case is explained as follow **[**34**]:**

Consider input $x_i = \{x_1, \ldots, x_n\}$ where n is the total number of samples, is belong to with class 1 (y_i = 1) and class 2 (y_i = -1). For linear separable data, hyper-plane that separates data is found so that decision function $f(x)$ in Eq. (2.4) is equal to zero.

$$f(x) = \omega \cdot x + b = \sum_{i=1}^{n} \omega_i x_i + b = 0 \qquad (2.4)$$

where ω is an n-dimensional vector and b is a scalar.

Position of the separating hyper-plane is determined by vector ω and scalar b. A distinctly separating hyper-plane satisfies the constraints in Eq. (2.5). Optimal separating hyper-plane is determined from hyper-plane that creates maximum margin.

$$y_i(x_i \cdot \omega + \mathrm{b}) \geq 0 \Leftrightarrow \begin{cases} f(x_i) = x_i \cdot \omega + \mathrm{b} \geq 1 & y_i = +1 \\ f(x_i) = x_i \cdot \omega + \mathrm{b} \geq 1 & y_i = -1 \end{cases} \qquad (2.5)$$

Other than linear separation data, SVM also shows its ability in handling nonlinear data. SVM classification is sensitive and has high accuracy. Survey done in **[**9**]** had shown that the accuracy could achieve up to 98% in finger vein recognition system. However, SVM only could achieve high accuracy performance with small training sample size and it is sensitive to noise **[**9**]**. Nevertheless, processing time performance is not improved in the reviewed works.

2.5.3 Sparse Representation classifier

In **[**14**] [**35**],** Sparse Representation classifier (SRC) has been proposed to be used in finger vein recognition system. SRC assumes information in a signal is linear combination of a small number of basic elements called atom **[**35**]**. Training samples is arranged as column vector in a dictionary matrix while testing sample is represented as another set of atom dictionary. Steps of classification using SRC is shown as follow **[**36**]**:

Step 1: Load training samples, $\boldsymbol{A} = \{\boldsymbol{A_1}, \ldots, \boldsymbol{A_k}\} \in R^{mxn}$ with k classes and a testing sample $\boldsymbol{b} \in R^m$, n = training sample index and m = data of the training sample.

Step 2: Normalize the columns of $\boldsymbol{A}$ to use unit l_2 norm

Step 3: Solve l_1 norm minimization problem in Eq. (2.6):

$$\boldsymbol{x}_1 = argmin_x \left||\boldsymbol{x}|\right|_1 \tag{2.6}$$

subjected to $\boldsymbol{b} = \boldsymbol{Ax}$ where $\boldsymbol{b}$ and $\boldsymbol{A}$ are as mentioned in step 1.

Step 4: Calculate the residuals, R_i for $i = 1, 2, \ldots, k$ using Eq.(2.7)

$$R_i(\boldsymbol{b}) = \|\mathbf{b} - \mathbf{A}\delta_i(\boldsymbol{x}_1)\|_2 \tag{2.7}$$

Step 5: Find identity of testing sample using Eq. (2.8)

$$\text{Identity } (\boldsymbol{b}) = \; argmin_x R_i(\boldsymbol{b}) \tag{2.8}$$

SRC is a powerful tool to for classifying large sample size and low dimensional sample **[**32**]**. However, SRC does not study similarity between testing and training sample. SRC also has high complexity because it could not obtain closed form solution using l_1 norm minimization and thus, it is time consuming **[**36**]**.

2.5.4 K-Nearest Neighbor

As survey in [9], in past few years, K-Nearest Neighbor (KNN) classifier is one of the popular classification methods to be used in finger vein recognition system. KNN is a non-parametric technique of classifying objects based on closest training samples in the sample space [37]. A query sample with unknown class is identified by finding its *k* nearest neighbors (NNs) from a set of training samples with defined class in database.

To choose NNs of the query sample, distance between the query sample, *x* and each training samples, x_i is measured. For a training set of N training samples, Euclidean distance (Eq. (2.9)) is used due to its simplicity [12] [17] [38].

$$Euclidean\ distance, E_i = ||x - x_i|| \tag{2.9}$$

where x_i = ith training sample, i = 1, 2, …, N.

The training sample with shortest distance with query sample is chosen as first NN (1-NN), second shortest distance with query sample is 2-NN and the assignment continues up to *k*-NN depend on *k* selected where *k* is the range of neighborhood which is also known as number of NN of query sample. After *k* NNs have been chosen, by putting the classes of those NNs in voting list, the query sample is assigned to the class that obtains majority votes. In example shown in Figure 2.5, query sample is assigned to class 2 as it gets three votes (dominant votes out of total five votes) at *k*=5.

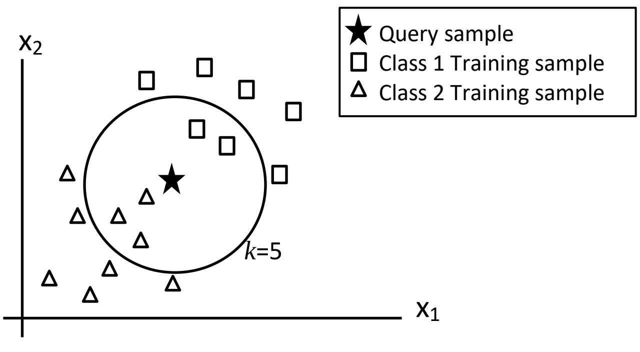


Figure 2.5: KNN classification at *k*=5

KNN classifier is more straightforward than other classifier and its processing time is also shorter. Main weakness of KNN classifier is low accuracy performance when deal with small training sample size. In [17], KNN has accuracy of 96.33% with processing time of 1.44 ms and SVM has accuracy of 96.83% with processing time of 23.93 ms. To improve this problem, LMKNN is proposed as improvement of KNN [20]. However, this classifier does not consider the weightage of each NN as well.

2.6 K- Nearest Centroid Neighbor

K-Nearest Centroid Neighbor (KNCN) classifier and its improved classifiers had often been applied in finger vein recognition [16] [16] [15]. KNCN is an extension of KNN that mentioned in section 2.5.5. Being advance than KNN, KNCN not only takes account of proximity of query sample but also consider its spatial distribution by finding its nearest centroid neighbors (NCN) instead of NN [19] [20]. Algorithm of KNCN classifier [39] is shown as follow.

For a set of training sample $T = \{(x_i, y_i)\}_{i=1}^{N}$ where x_i is ith training sample, y_i is class of ith training sample, i is training sample index and N is the total number of training samples in database, query sample is defined as x and its unknown class is represented as y.

Step 1: Set k- parameter to determine how many NCN to be found.

Step 2: Find 1-NCN (same as 1-NN) by calculating Euclidean distance between query sample and all training samples in set T using Eq. (2.9). Training sample with shortest distance is chosen as 1-NCN.Put the class of 1-NCN in voting list.

Step 3: Calculate the centroid of 1-NCN with each training sample. Centroid of a set of point $x = \{x_1, x_2, \dots, x_n\}$ can be calculated using Eq. (2.10)

$$c_n = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (2.10)$$

where c_n is centroid of *n* training samples and *n* is number of training sample involved in centroid calculation

Step 4: Find next NCN by choosing the training sample that has shortest Euclidean distance between the query sample and centroid found in step 3. Training samples that already been chosen as NCN are ignored.

Step 5: Put the class of that training sample in voting list.

Step 6: Recalculate the centroid of each training sample by adding the new NCN found in step 4 to the calculation in step 3.

Step 7: Repeat step 4, 5 and 6 to find the rest of NCN until *k*-NCN

Step 8: Carry out class voting process. Class of the query sample will be assigned to the class that obtain majority votes during voting process

KNCN employs concept of centroid of between query sample and NCN as reference point instead of the query sample itself. Comparison between NCN and NN is shown in Figure 2.6.

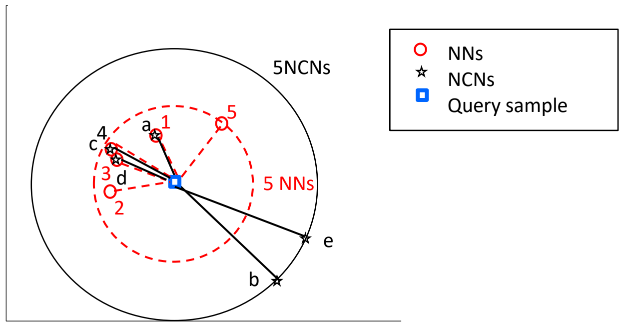


Figure 2.6: Comparison between NN and NCN classifier at *k*=5 [37]

As shown in Figure 2.6, at same value of *k*, KNCN classifier has wider neighborhood distribution than KNN classifier. This is due to the changing of centroid for calculation of every added NCN. In other words, query sample has more possibility to be classified correctly if it is close to its centroid **[**19**]**. This is because samples that belong to same class normally appear as a cluster. With the concept of centroid, in KNCN classifier, distance between classes is enlarging while distances within class are narrowing.

However, some problems arise when centroid concept is used. This is due to neighbors corresponding to the nearest centroid might not nearest to the query point, hence, might not belong to the same class. Centroids might approach to the query sample although the NCN is getting far as value of *k* increases. Other problems that exist in KNN classifier, such as ineffectiveness when deal with small training samples and assumption of all NNs (NCNs for KNCN classifier) has same weight in voting process, also exist in KNCN classifier This causes inaccuracy in classification process in finger vein identification system **[**19**]**.

2.6.1 Local Mean K-Nearest Centroid Neighbor

To overcome problem of small training size in KNCN, LMKNCN **[**40**]** is proposed. Procedure to develop LMKNCN is stated in the following steps **[**20**]**:

Let $T = \{x_n \in \Re^m\}_{n=1}^{N}$ be a set of training sample with *M* classes $c_1, \dots, c_M$ and $T_i^{NCN}(x) = \left\{x_{ij}^{NCN} \in \Re^m\right\}_{j=1}^{N_i}$ be class of training set of c_i, each of which consists of N_i training samples.

Step 1: Find set of k NCNs from the set T_i of each class c_i for the query pattern x.

Step 2: Compute the local centroid mean vector u_{ik}^{NCN} using Eq. (2.11) from each class c_i using the set $T_{ik}^{NCN}(x)$

$$u_{ik}^{NCN} = \frac{1}{k}\sum_{j=1}^{k} x_{ij}^{NCN} \tag{2.11}$$

Step 3: Calculate distance $d\left(x, u_{ik}^{NCN}\right)$ between x and the local centroid mean vector u_{ik}^{NCN} using equation Eq. (2.9).

Step 4: Assign x to the class c, which has the closest distance between its local centroid mean vector and the query pattern x by using Eq. (2.12)

$$c = \arg min_{c_i} d(\ x, u_{ik}^{NCN}) \tag{2.12}$$

In **[**20**]**, it is proven that LMKNCN has better accuracy (99.15%) than KNCN (99.08%) and KNN (98.71%) because it not only considers geometric distribution of testing sample but also solves the problem of low accuracy due to small training sample size . However, this method does not consider the weightage of each NCN during voting process to classify a query sample.

2.6.2 Weighted K-Nearest Centroid Neighbor

WKNCN improves the local mean vector of *k* neighbors from each class in making classification decision based on weight voting scheme. Two weight voting method **[**18**]** are suggested using formula Eq. (2.13) and Eq. (2.14) where width of kernel, *t* in Eq. (2.14) is calculated using Eq. (2.15)

$$Uniform\ kernel\ function: w_i^{NCN} = \frac{1}{i}, i = 1, 2, \dots, k \tag{2.13}$$

$$Heat\ kernel\ function:\ w_i^{NCN} = exp\left(-\frac{\left\|x-x_i^{NCN}\right\|^2}{t}\right), i = 1, \dots, k \tag{2.14}$$

$$t = \frac{1}{k^2}\sum_{i=1}^{k}\left\|x - x_i^{NCN}\right\|^2 \tag{2.15}$$

where i is the position of k nearest centroid neighbour.

After selecting k-NCN, the weight of each NCN is computed using either Eq. (2.13) or Eq. (2.14), and the class of query sample, y is decided using Eq. (2.16)

$$y = \arg max_{c_j} \sum_{(x_i^{NCN}, y) \in T_k^{NCN}(x)} w_i^{NCN} * \ \delta(c_j = y_i^{NCN}) \tag{2.16}$$

where $\delta(c_j = y_i^{NCN})$ outputs a value of one if $c_j = y_i^{NCN}$, and zero otherwise.

Experiment in **[**40**]** has proven that WKNCN performs better than LMKNCN in classification. However, work is not done to improve processing time performance of the classifier in that paper.

2.7 Summary

Finger vein recognition has certain strengths over other biometric techniques and it is worth to have further research. In this chapter, general model of finger vein recognition is studied as a guideline for project implementation. Several classifiers that have been proposed previously for finger vein recognition are reviewed and strengths and weaknesses of the reviewed classifiers are summarized in Table 2.1. However, almost all the reviewed work only focus in accuracy performance in finger vein recognition without considers its processing time performance. In order to make KNCN classifier to be more preferable in finger vein recognition system, improvements need to be done on it based on the problem found.

Table 2.1: Summary of strengths and weaknesses of classifiers

Classifier	Accuracy (%)	Database	Strengths	Weaknesses
Naïve Bayes	91.00 [12]	Private database	1. Low complexity 2. Perform well in noisy environment.	1. Less practical to deal with real-world data
SVM	99.64 [25]	SDUMLA-HMT	1. Can handle nonlinear data 2. High accuracy	1. Only perform well with small sample size 2. Sensitive to noise
SRC	99.98 [35]	Private database	1. High accuracy for large sample size and low dimensional sample	1. High complexity 2. Do not study similarity between testing and training samples.
KNN	77.03 [16]	FV- USM	1. Low complexity	1. Low accuracy for small training sample size 2. Neglect weightage of each NN in voting process
	98.6 [18]	Private database		
	98.53 [21]	Private database		
KNCN	78.64 [16]	FV-USM	1. Higher accuracy than KNN due to consider the proximity and spatial distribution 2. Low complexity	1. Same as weaknesses in KNN 2. Long processing time for large value of K 3. Overestimate range of training sample to be chosen as NCN
LMKNCN	100.00 [21]	Private database	1. High accuracy for all sample size	1. Neglect weightage of NCNs during voting process 2. Long processing time for large value of K
WKNCN	99.70 [18]	Private database	1. High accuracy 2. Consider weightage of NCNs in voting process	1. Long processing time for large value of K

CHAPTER 3

METHODOLOGY

3.1 Overview

This chapter describes the methodology of implementation of project with the aim to propose an improved KNCN classifier for finger vein recognition. Overall project implementation flow and project requirement are explained in section 3.2 of this chapter.

The whole project design is presented in section 3.3. Beginning of this section discusses the development of typical KNCN classifier and analysis on it to obtain information to improve the classifier. Next, method to improve accuracy of the classifier and experiment on the improved classifier is explained in detail. Last part of this section shows method to modify the improved classifier to reduce its processing time.

In section 3.4, feature extraction (another process in finger vein recognition system that is included in this project) is described. Purpose of this process and the method used is stated in this section.

Next, method to evaluate performance of proposed classifiers is stated in section 3.5. Database used for this project is also introduced in this section. Finally, this chapter is summarized in section 3.6.

3.2 Project Implementation Flow

Before this project starts, the overall project flow is drawn as a guideline to develop the project in order to achieve objectives targeted. The flow chart of the overall project implementation is illustrated in Figure 3.1.

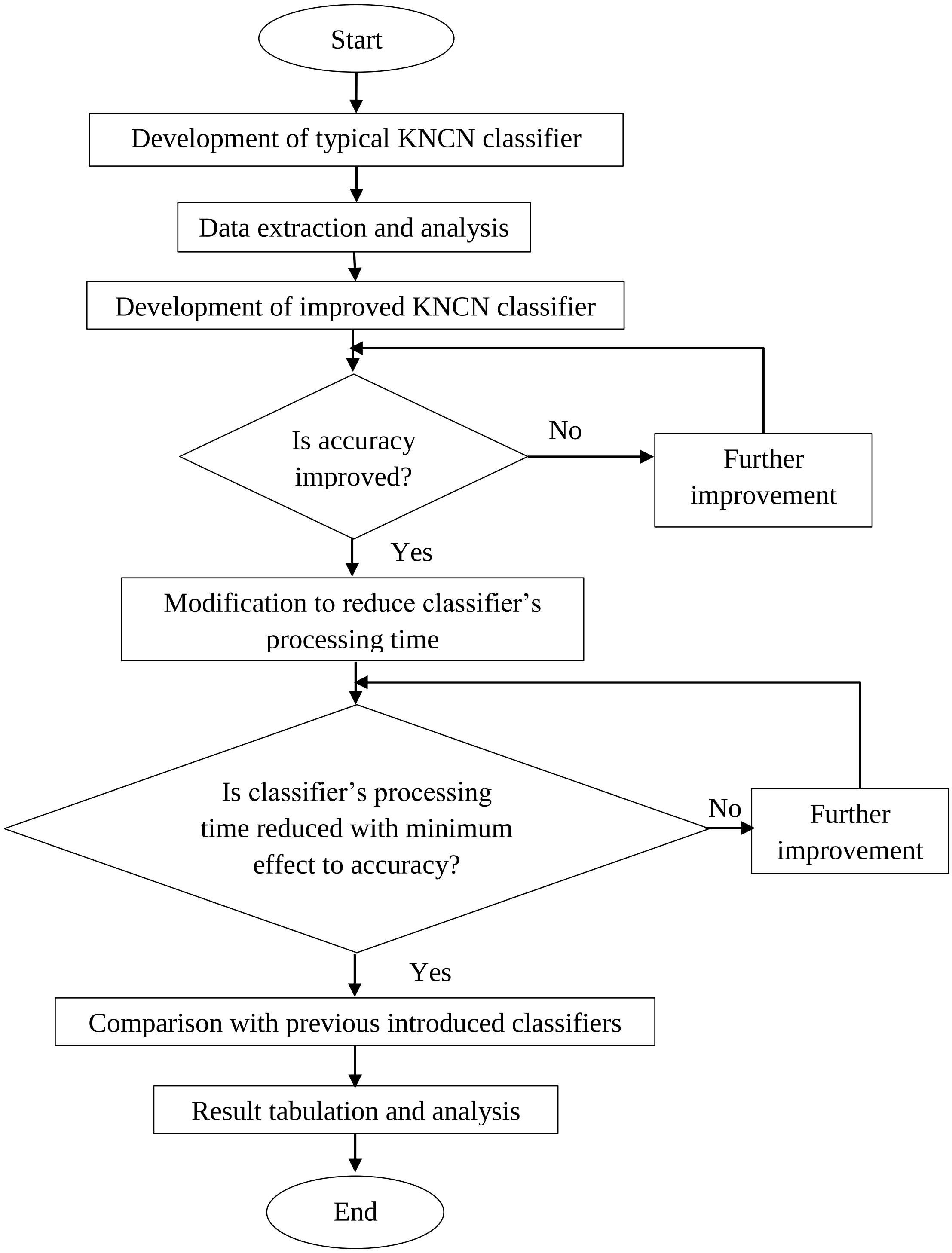


Figure 3.1: Overall project flow chart