Secure Network Authentication Based on Biometric National Identification Number

A Thesis

Submitted to the College of Information Engineering

at Al-Nahrain University in Partial Fulfillment of the

Requirements for the Degree of Master of Science

in

Networks Engineering and Internet Technologies

by

Muntasser Saleem Falih

(B.Sc. Information and Communication Engineering 2013)

Rabi Al-Thani **1437**

February **2016**

Supervisor Certification

I certify that this thesis entitled "**Secure Network Authentication Based on Biometric National Identification Number**" was prepared by "**Muntasser Saleem Falih**" under my supervision at Networks Engineering Department/ College of Information Engineering / Al-Nahrain University in partial fulfillment of the requirements for the degree of **Master of Science in Networks Engineering and Internet Technologies.**

Supervisor

Assist. Prof. Dr. Mahmood K. Ibrahem

/ / 2016

In view of the available recommendation, I forward this thesis for debate by the Examining Committee.

Assist. Prof. Dr. Emad H. Al-Hemiary

Head of Networks Engineering Department

/ / 2016

Committee Certificate

We the Examining Committee, after reading this thesis entitled " **Secure Network Authentication Based on Biometric National Identification Number** " and examining the student "**Muntasser Saleem Falih**" in its content, find it is adequate as a thesis for the degree of **Master of Science in Networks Engineering and Internet Technologies.**

Prof. Dr. Ahmed T. Sadiq
Chairman
/ / 2016

Assist. Prof. Dr. Haithem A. Omer
Member
/ / 2016

Dr. Mohammed F. Abdul Kareem
Member
/ / 2016

Assist. Prof. Dr. Mahmood K. Ibrahem
Supervisor and Member
/ / 2016

Approved by the College of Information Engineering/ Al-Nahrain University.

Prof. Dr. Mohammed Z. Al-Faiz
Dean, College of Information Engineering/ Al-Nahrain University

/ / 2016

Abstract

Networks authentication is a very important security issue. The classical authentication methods (token-based and knowledge-based) suffer from drawbacks like; stealing, forgotten and guessing, these drawbacks lead to the use of “biometric authentication”.

Fingerprint is one of the important physiological biometric types which has sufficient uniqueness to distinguish among persons and led to use in several security applications.

This research present a proposed authentication system based on fingerprint as biometric type and some of static credential personal information such as name and birth date. The system generates a Unique National Identification Number (NIDN) by combining fingerprint minutiae features with user’s personal information (name and birthdate) printed in Quick Response code (QR) image used as a token card for accessing the system. One-to-one fingerprint verification is used to verify user’s identity for the application with high security level.

The proposed system provides two authentication services: The first is normal authentication service to protect the public application that contains public data such as billing payments application. This authentication service needs only the user’s QR token for system access. The second service is strong authentication for protecting private application that contains sensitive data such as banking. This authentication service needs user’s QR token and fingerprint for system access.

The experimental work of the proposed system shows that, with threshold value under (50), user's acceptance accuracy is 100%, and with threshold value equal to (50), user's acceptance accuracy is 96.153. Incrementing the threshold leads to increase in user's rejection rate by dropping the less matching experiments and keeping the full match experiments. High threshold value (>50) can be used with high level security applications.

List of Contents

List of Abbreviations

Abbreviation	Meaning
AFIS	Automatic Fingerprint Identification System
ATM	Automatic Teller Machine
BMP	Bitmap
CN	Crossing Number
DB	Database
FAR	False Acceptance Rate
FIM	Federation Identity Management
FP	Fingerprint
FRR	False Rejection Rate
GF	Gabor Filter
GUI	Graphical User Interface
HTTP	Hyper Text Transfer Protocol
IP	Internet Protocol
LAN	Local Area Network
MD5	Message Digest Five
NIDN	National Identification Number
QR	Quick Response
SDK	Software Development Kit
SSL	Secure Socket Layer
SSO	Single Sign On
TCP/IP	Transmission Control Protocol/Internet protocol
UIDM	Unique Identity Management

List of Symbols

Symbol	Meaning
M	Mean value
V	Variance
N	Normalized grey-level
I	Grey-level value
(i,j)	Pixel position
W	Block size value
∂_x	Horizontal gradient magnitude
∂_y	Vertical gradient magnitude
$\theta(i,j)$	Least square estimate of local orientation
Φ_x	Vector field component in x-axis
Φ_y	Vector field component in y-axis
G	Gaussian filter
$O(i,j)$	Orientation image
$F(i,j)$	Local ridge frequency
$S(i,j)$	Local ridge spacing
$E(i,j)$	Gabor filter transfer function
K	Check point pixel
A_i	Neighboring pixels

List of Tables

List of Figures

Chapter One

Introduction

1.1 Motivation

The presence of modern sensitive applications such as e-government, banking transaction and smart card and assurance on the protection of the information saved in multiple Databases (DB). Automatic personal identification become most important issue, broad range of civilian applications need accurate automatic personal identification.

Personal identification is the process of binding a specific individual for an identity. Identification sometimes came in the form of verification which also refer to (authentication) or (recognition), this means defining a user identity from DB of users known to the system. Two personal identification techniques have wide range of usage are Token-based and Knowledge-based. Token-based techniques use of “something you have” to generate a personal identification, examples on these technique such as passport, ID card, driver’s license, credit card or keys. Knowledge-based technique uses of “something you know” to generate a personal identification, examples on these techniques Password or Personal Identification Number (PIN). The two techniques have some drawbacks, lost, stealing, forgotten or misplaced are associated with a token and guessing, forgetting are associated with a password or PIN. A more reliable and secure approach can be used to support person’s identity instead of the traditional approaches is called “Biometrics” [1].

Biometric is technology that helps to make data be more secure. It defines all users by the way of their personal physical or behavioral properties. Fingerprint (FP), face, iris, speech, handwriting or hand geometry and so on are the biometrics information that can be used to completely identify people. Using biometric identifiers present several advantages over other identifiers (token and knowledge based) [2].

From all biometrics types, FP has one of the highest security and authenticity levels [3]. FP is the best biometrics to be easily captured, stored and compared to verify the identity of an individual. A FP is an active proof of a person's identity as a part of the fingerprint uniqueness and universality [3, 4].

This research (Secure Network authentication Based on Biometric National Identification Number) focuses on FP as biometric technique to provide the authentication for each user accessing an e-government system. FP features are unique for each individual, this features can be combined with some credential personal information such as Name and Birthday date, to produce a unique National Identification Number (NIDN). This generated number can be used in authentication process as identification number, as well as FP recognition (verification) technique can be used to verify the user identity.

1.2 Literature survey

Many researchers have been introduced advances to FP image processing, features extraction, code generation from FP features and FP recognitions.

1.2.1 Fingerprint image enhancement and features extraction

L. Hong, Y. Wan and A. Jain, in 1998[5], they have presented a fast FP enhancement algorithm, which can adaptively improve the clarity of ridge and valley of input FP image, this algorithm based on the estimation of local ridges orientation and frequency.

R. Thai, in 2003[6], discussed the role of FP in identification systems and explained the statistical theory of FP minutiae to reliably extract minutiae from FP image by providing framework to the enhancement and minutiae extraction of FP image. Experiments using a mixture of both artificial test images and real FP images are then conducted to evaluate the performance of implemented technique.

I. Babatunde, A. Kayode, A. Charles and O. Olatubosun, in 2012 [7], have presented a modification to a sub–models of an existing mathematical FP image enhancement algorithm to obtain a new and improved version. The new version consist of different mathematical models for FP image segmentation, normalization, ridge orientation estimation, ridge frequency estimation, Gabor Filter (GF), binarization and thining. The results show that the modified sub-models perform well with significant improvement over the original versions.

N. Negi and S. Semwal, in 2013 [8], have presented a developed algorithm for FP minutiae extraction with varying quality, preprocessing in the form of filtering, binarization and thinning is first applied on the FP image before they evaluated. Improved FP pattern generated by a developed features pruning algorithm and then classification is done by the radial basis function network.

1.2.2 Code generation from fingerprint features

B. Ne'ma and H. Ali, in 2009[9], have presented a method for multi-purpose code generation, this code can be used in different security application such as ATM, coded door locks and other security measures. The method of generation this code consists from two phases, the first phase dealing with FP enhancement and thinning, the second phase dealing with feature (minutiae) ridge and bifurcation extraction. Finally the secure code is generated by applying the one-way hashing function Message Digest Five (MD5) to the extracted feature.

I. Jabber, in 2012 [10], have presented an algorithm to generate a number of hashes based on a FP technology. These hashes can be used for user's identification and authentication processes. The local and global features have been used a robust recognition system where a robust algorithm is used to extract minutiae features accurately.

H. Salman, in 2013 [11], have presented a design of Fingerprint Random Number Generator (FPRNG), which produce non repeated, endless number. The location of minutiae on FP image is used as a seed for random number generation were carried out using residue classes and the complete system of residue classes modulo (n) as mathematical model.

S. Ambadiyil, K. Soorej and V. Pillai, in 2015 [12], proposed a method for creating a unique Identification Number (ID) based on FP core point and minutiae features. In these research minutiae features are extracted with core point as reference point, the block of these features and core point are included in Quick Reader Code (QR) and printed in security document.

1.2.3 Fingerprint recognitions

P. Zhang, X. Guo and J. Gadedadikar, in 2011 [13], have presented a framework on the FP verification distributed system based on web environment, in this work the process begin form FP acquisition, FP image processing, feature extraction this done in client side. The features transmitted via internet to server side. The server process or verify this feature with template which is already saved in the server's database, then server replay the verification result to client side.

Y. Li-qiang and G. Ling, in 2012 [14], have presented an algorithm for minutiae extraction depend on finding the thin ridge line, showing the type of current point and the states of its 8-neighborhood pixels by 8-neighbour coding, which can perfectly extract ridge endings and ridge bifurcations in thinned. Blocking of the faked minutiae is achieved by this algorithm, the speed of features extraction also enhanced to satisfy the need in the practical application. All this is showing in the experimental results.

N. Bhargava, R. Bhargava, M. Mathuria and M. Cotia, in 2012 [15], have presented a matching algorithm which find the correspondence between two FP based on ridge ending and bifurcation points. The FP matching accomplished. This algorithm performs two operations, first, to calculate the available points on the FP surface and second, to find out location of these points, then make comparison between these point's data of two FP impressions.

1.3 Aim of work

The aim of this project is to design and implement a secure network authentication system by generating unique National Identification Number (NIDN) based on fingerprint (FP) minutiae features as a type of biometrics mixed with some of static credential personal information such as name and birthday date. This NIDN can be used as identification number in networking access by including this number in Quick Response code (QR) to be used as a token card instead of using expensive smart card. Using FP matching algorithm to provide verification process based on the personal FP. Provide secure data exchanging by using Secure Socket Layer Protocol (SSL). This generated system can be employed in networking security for the user's authentication in public and private applications.

1.4 Thesis Outline

Thesis strategy started in chapter one with introduction of personal identification, biometrics and FP, and related work that associated to FP processing and biometric ID generation. Chapter two presents brief background for biometric identification, FP acquisition, enhancement, features extraction and matching. Also, federation identity management, unique identity management, hashing function, QR and security protocol for client server interaction are described in this chapter. Chapter Three presents detail authentication system design and its building blocks. Chapter Four presents implementation of the proposed system, the execution results also presented in this chapter. Conclusion and suggestions for future work provided in Chapter Five.

Chapter Two

Biometric Identification

2.1 Introduction

This chapter provides an introduction to the biometric definition, biometric identification and its techniques, FP technique as type of biometric which is used in this research with detailed explanation about FP characteristics. A brief description to the Automatic Fingerprint Identification System (AFIS) with its activity like: FP image enhancement, features extraction algorithm and FP matching techniques. Finally this chapter gives brief description to the identity management, Federation Identity Management (FIM), Unique Identification System, QR code architecture, Hashing Function and SSL protocol that used in the networking security for secure data streaming form.

2.2 Biometric Definition

Biometric is the science of finding an individual identity from the physical or behavioral characteristics of the person. The expression biometric is composed of two Greek term “bio” for life and “metros” for metric [16]. Physiological and behavioral attributes of any human can be selected as person identifiers if satisfies some of requirements [17].

2.2.1 Theoretical requirements:

1. **Universality:** according to this requirement, each person must have biometrics features.

2. **Distinctiveness:** means that biometrics of any two persons must be sufficiently unique to distinguishing between them.
3. **Permanence:** means that biometrics features of any person must be static (not change) along his life.
4. **Collectability:** this means that the selected biometrics must be quantitatively measured.

2.2.2 Practical requirements:

1. **Performance:** refers to the accuracy, speed and hardness of the recognition system.
2. **Acceptability:** refers to the range of people acceptability for any biometrics characteristics to be daily used in their identification.
3. **Circumvention:** refers to the tolerance in fooling the system by faked techniques.

2.3 Biometric Identification

A biometric identification system is basically a pattern recognition system that performs the recognition depend on some features extracted from measurement of physiological or behavioral characteristics. Biometrics attributes includes; FP, face, Iris, geometry form of hand, gait, voice, keystroke and signature [18]. Figure 2-1 illustrates types of biometric identification.

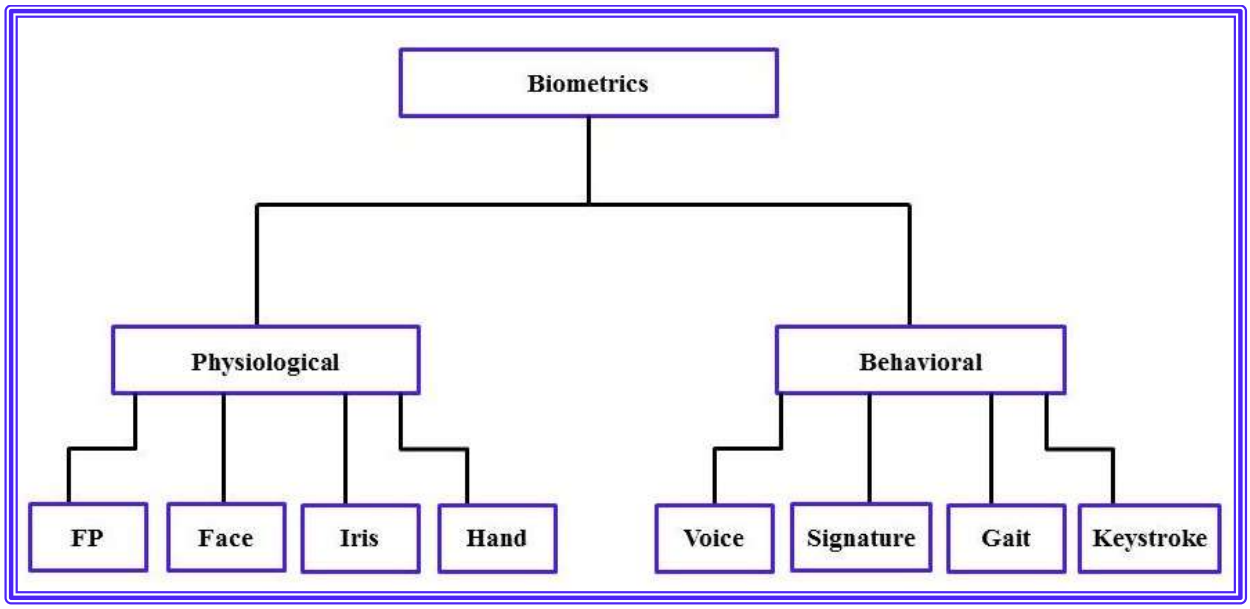

Figure 2-1: Biometric identification attributes

2.3.1 Fingerprint

Fingerprint (FP) is one types of biometric attributes that has wide range of authentication applications such as national ID card, airport check-in, border control, driver's-license authenticity, computer network login, physical access control, electronic banking, personal authentication,… etc. The reason for the important of fingerprint related to its' uniqueness, stability for long life and existence of capturing scanners. The recognition system of fingerprint system consist of five modules; image acquisition, fingerprint image enhancement, features extraction, store features and matching input image with store image in Data Base (template) [19]. FP identification has been used in this research and will be discussed in more details later on.

2.3.2 Face recognition

Face recognition is an evident way and facial components may be the most popular biometric features utilized by humans to recognize one another. Face recognition have wide range of applications beginning from a constant, controlled "mug-shot" authentication to vital, uncontrolled face identification in a confusion background. There are number of face recognition method; location and shape of facial components method based such as eyes, nose, lips, eyebrows and chin, the second method is total analysis of face image that define a face as a weight set of a number of accepted faces [20].

2.3.3 Hand geometry

Human hand can be measured to give a certain features used in hand geometry identification system, these measuring features include; form, size of palm, and the dimension of fingers (length and width) [21]. In application environment, hand geometry identification system has wide range of usage. Simplicity, easy in use and cheapness are characteristics related to this biometric technology. Recognition accuracy of the hand geometry is not affected by some environment factors like dry weather or dry skin.

2.3.4 Iris

Iris is one type of physiological biometrics located in the region between the pupil and the sclera (white of the eye) in human eye is called. Biometric identification of person based on human iris is fits to be applying in access control and provide trust security. Comparative to the others biometric features, iris is a more reliable and accurate for authentication process [22].

2.3.5 Voice

As behavioral biometric the voice recognition (voice print) is as unique features to an individual as others features, fingerprint or hand geometry. The verification process of voice recognition is depends on different characteristics of person's voice to distinguish between speakers [23]. Text dependent and text independent are two ways to achieve the voice recognition system. In text dependent method, user authentication is based on user's speech for some fixed predefined phrase. In text independent method the recognition way is more complex but offers high levels of security and toward the faked access [21].

2.3.6 Signature

Name signing way is called "signature", it define as one type of behavioral biometric type. Signature needs some user's requirements such as contacting with writing tool and personal effort. Signature has been accepted as authentication way in business and government transactions. Online signature became as biometric option in modern device such as PDAs and Tablet PCs. The different in signature pattern in each experiment considers as weak point gives the chance to fool signature authentication system by fabricated access [21].

2.3.7 Gait

Gait means the pattern of how people are walking, and is one of biometrics attributes that has little usage in people recognition at a distance. Gait recognition can be used in secret monitoring of individuals, as well as gait recognition provides individuals tracking for a period of time. Gait recognition system algorithms used an extracted point of moving which is represents by optic flow to describe the gait of an individual. Finally, some of impact factors

can be influenced on the gait recognition system performance such as choice of shoe, nature of land surface, relationship of legs and clothing type [24].

2.3.8 Keystroke

As a behavioral biometric, keystroke proposed that each person has a typing cadence on a keyboard in a distinctive form. Keystroke in not predicted to be unique to each individual but it may be predicted to provide suitable special information to allow verification [25]. Behavioral biometric systems including keystroke as a way of authentication has advantage of capturing even without the knowledge of the use. System has software to record the keystroke and timing information, can be used to classify users based on their behavior in typing [26].

Figure 2-2 illustrates different biometric techniques.

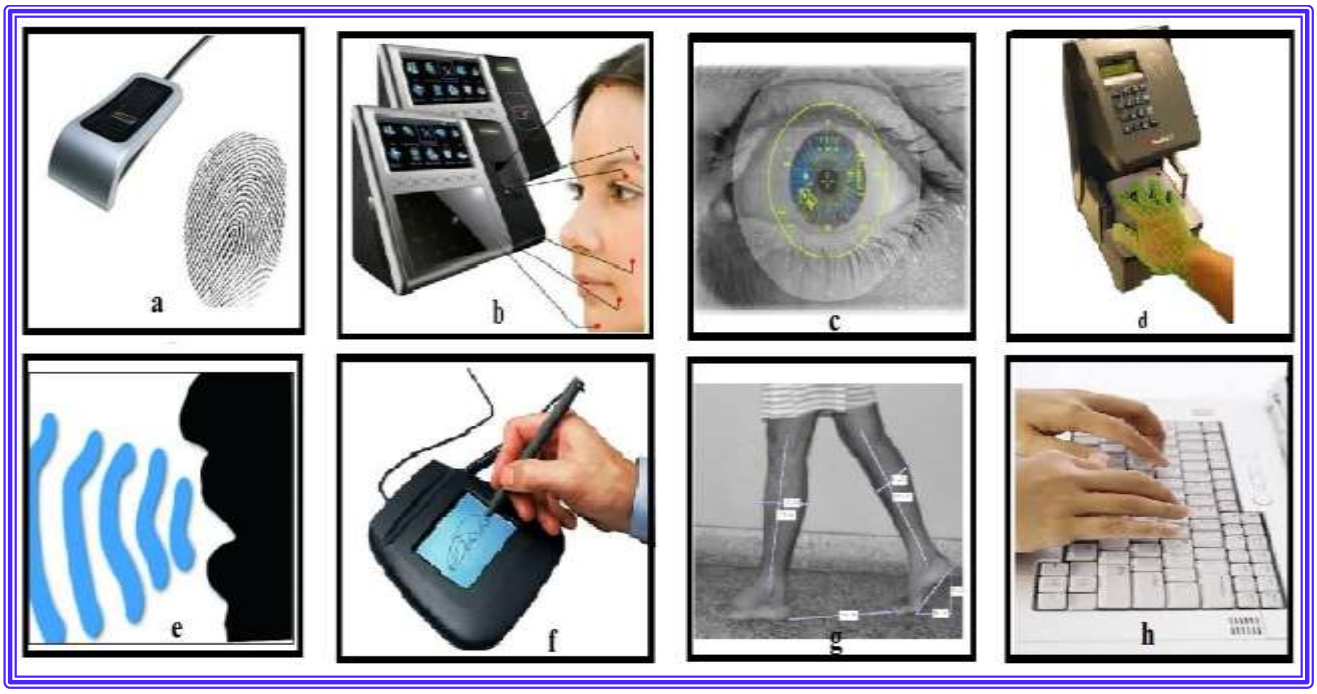

Figure 2-2: Biometric technique; (a) FP, (b) face, (c) iris, (d) hand geometry, (e) voice, (f) signature, (g) gait, and (h) keystroke**.**

2.4 FP in More Details

FP is an arrangement of patterns consists from ridges and valley on the surface of human finger. In FP image, ridges are dark, whereas valleys are bright [27]. FP identification relies on the stability and individuality of FP. Stability refers to the permanent and not changing the pattern of each finger from birth until death. Individuality indicates the uniqueness of ridge details toward individuals [28]. Figure 2-3 reveals the FP image.

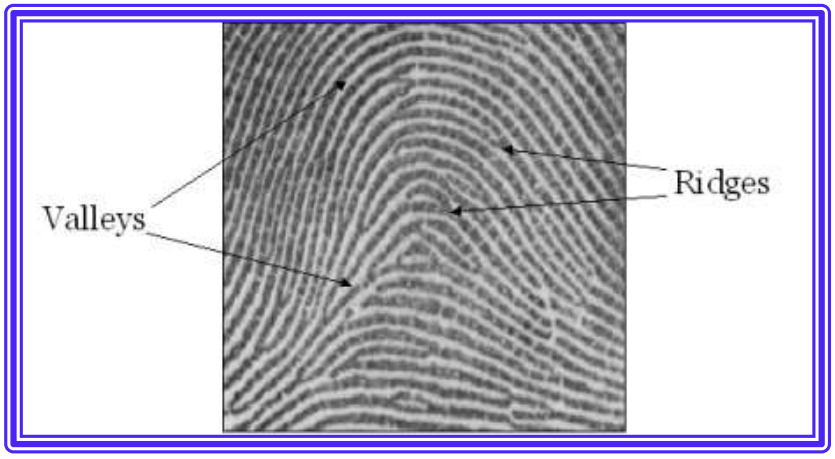

Figure 2-3: FP image

2.4.1 FP Grouping

In the grouping of FPs, there are many of basic pattern ridges classes. The most common three types are; loop, arch and whorl. The loop is the most popular pattern type of FPs. Figure 2-4 (a) reveals the loop pattern. Statistical measurement for about 65% of all classification of FPs is returned to the loop [29, 30]. The arch type is different from loop it has more open curve. Arch patterns can be classified into two types; Plain Arch and Tented Arch as shown

in Figure 2-4 (b). The whorl pattern appear as at least one ridge that creates a complete circle, statistically occurrence of whorl is about 30% of all FPs, Figure 2-2 (c) illustrates whorl pattern [30].

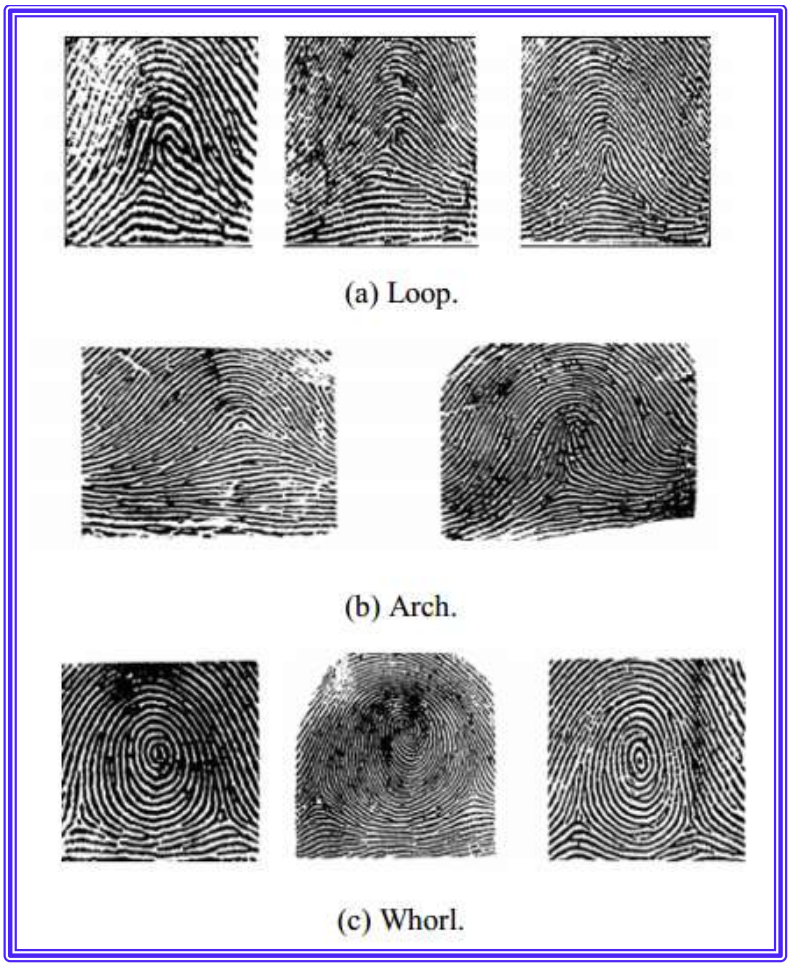

Figure 2-4: FP ridges patterns classification [9]

2.4.2 FP features

There are two types of features that can be extracted from FP images:

1. **Global features:** which characterize the FP pattern class. The important global features are the core and delta regions. For example a loop pattern

has one core and one delta regions, a whorl pattern has one core and two delta regions, and an arch pattern has neither a core nor a delta region [31]. Figure 2-5 illustrates FP global features.

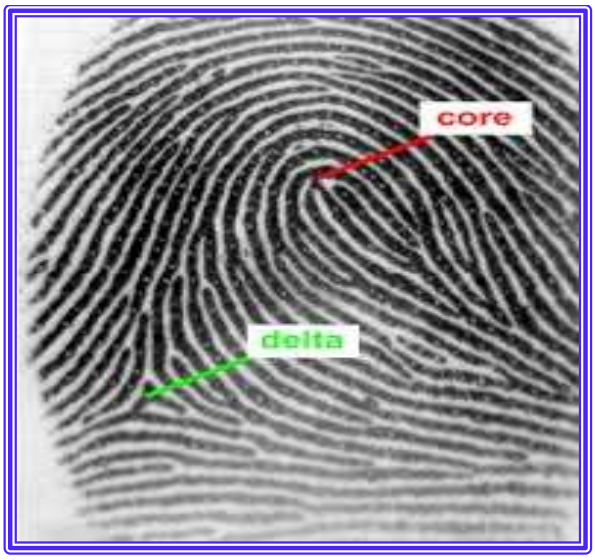

Figure 2-5: FP global features (core and delta points)

2. **The low level features:** are the ridge characteristics which called minutiae. Points of ridge ending (terminations) and bifurcations (branching), are unique features for each FP image and can be detected from thinned FP image [31, 17]. Figure. 2-6 reveals ridge ending and bifurcations.

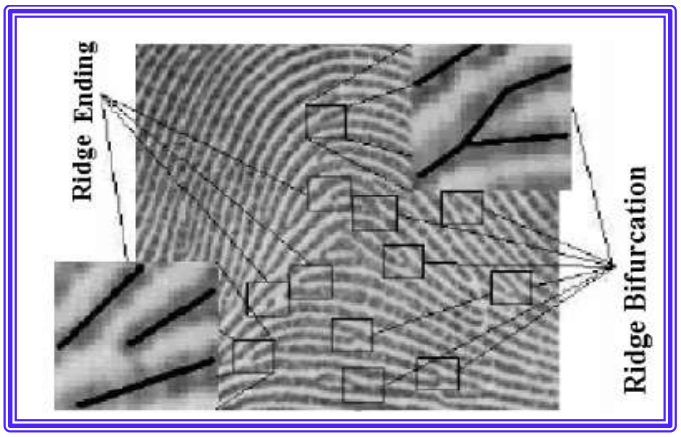

Figure 2-6: ridge ending and bifurcations (minutiae) [9]

2.4.3 FP Sensing

Acquisition mode of FP image can be classified either off-line, or live scan. The off-line mode represented by classical ink impression of FP on the paper, these inked impression can be converted to digital FP image by scanning paper using digital scanner or high resolution video camera. The live-scan mode generate digital FP image by using special scanner by contacting the tip of finger on the scanner surface. There are number of live-scan sensing mechanisms (e.g., optical Frustrated Total Internal Reflection (FTIR), capacitive, thermal, pressure-based, ultrasound… etc.). Table 2-1 gives examples on FP scanners [27]. The resolution of FP image is measured by number of dots or pixels per inch (dpi). A resolution of 500 dpi is a minimum quality can be accepted by Federal Bureau of Investigation (FBI).

Table 2-1: FP sensors specifications

	Technology	**Company**	**Model**	**Dpi**
Optical-state	FTIR	Biomrtrika	Hiscan	500
	FTIR	Crossmatch	Verifier 300	500
	FTIR	Digitalpersona	UaraU4000	512
	FTIR	L-1 identity	DFR	500
	FTIR	Sagem	MSO350	500
	FTIR	Secugen	Hamster IV	500
Solid-state	Capacitive	Upek	Touch Chip	508
	Thermal	Atmel	Finger Chip	500
	Electric field	Authentic	AES4000	250
	Piezoelectric	BMF	BLP-100	406

2.5 Automatic Fingerprint Identification System

An Automatic Fingerprint Identification System (AFIS), can be defined as a computer-based system used to compare large number of FPs. AFIS is commonly used by forensic science to support criminal investigation because AFIS save more time in matching process than the classical techniques that operate by an expert person who matches the FPs manually.

2.5.1 AFIS Challenges

Automatic Fingerprint Identification System confronted some of challenges, this challenges factors make the AFIS unstable and low accurate, these sensitive factors are associated with the FPs images, example on these factors are [32]:

1. FP image rotation
2. FP image scaling
3. Noise
4. Large intra-class variations (variations in FP images of the same finger)
5. Large interclass similarity (similarity between FP images from different fingers)

2.6 AFIS Modules

AFIS consist from many stages starts with loading FP image and ends with returning matching score [32]. Figure 2-7 illustrates AFIS modules.

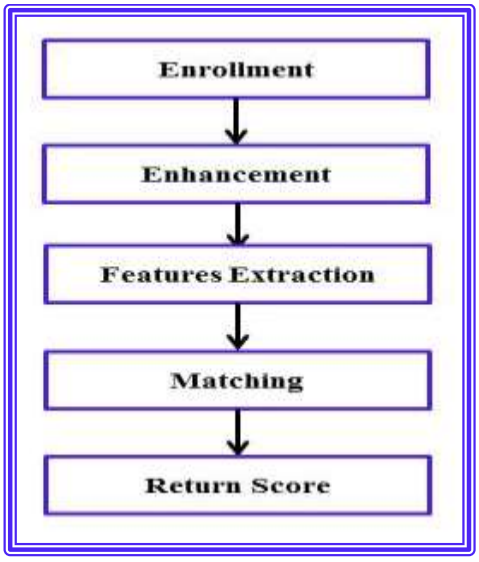

Figure 2-7: AFIS modules

2.6.1 Enrollment

The first stage of the AFIS is enrollment process. In this phase the user of the system will input his/her FP to be saved in database as template.

2.6.2 FP enhancement

FP image is rarely of perfect quality, the noise factors which are generated by some factors such as variation in skin, impression condition and sensing device. They may degrade and corrupted FP image. This degradation and corruption led to create new fake minutiae or important minutiae may ignore. Thus, it is necessary to use FP enhancement algorithms [33].

There are three main methods for FP image enhancement:

1. **Pixel-wise method:** in this method provides new value of each pixel depends only on the previous value of same pixel and some global parameter (not on the value of the neighborhood pixels) [5, 34].
2. **Contextual filtering:** enhancement in this method achieved by changing filter characteristics according to FP image local context such as local ridge frequency and local ridge orientation [35, 36].
3. **Multi-resolution method:** in this method multi-resolution approach is used to remove noise from FP image, like wavelet [37, 38].

The enhancement of any FP image can be done in; spatial domain, frequency domain and fuzzy domain [39]

The quality of FP image is very important because FP is rarely good quality, so FP image must enhance to clear its ridge and simplify minutiae

extraction with minimum error. In this research Gabor Filter (GF) approach has been used to enhance FP image quality.

GF method for FP image enhancement is widely used for FP application such as classification and matching. GF is a band-pass filter works in frequency and orientation selection, this means the filter can be tuned according to frequency and orientation values of the FP ridges. This algorithm is proposed by Raymond Thai which gives good results [6]. This algorithm consists from number of stages are: segmentation, normalization, orientation estimation, frequency estimation and GF [6]. Figure 2-8 shows the FP enhancement algorithm stages.

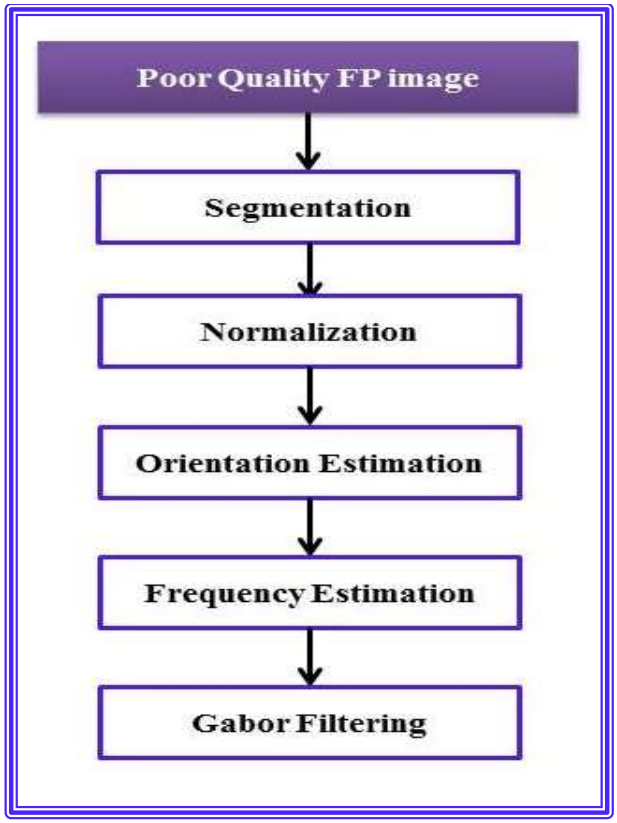

Figure 2-8: Enhancement algorithm stages.

a. **Segmentation:** is the process of separating the foreground regions from background regions in FP image. The foreground regions represent the ridges and valleys of FP image so the foreground is more important than background because it represents the region of interest. Foreground regions are higher grey-levels variance than background regions which is low grey-levels variance, so the method used for segmentation is based on the variance thresholding as follows [6]:

$$V(K) = \frac{1}{w^2} \sum_{i=0}^{w-1} \sum_{j=0}^{w-1} (I(i,j) - M(K))^2 \quad (2.1)$$

where V(k) is variance of block size W × W, $I(i,j)$ is grey-level value at pixel (i,j) and M(k) is mean value of block size W × W. Mean value M(K) is calculated by the following equation:

$$M(K) = \frac{1}{w^2} \sum_{i=0}^{w-1} \sum_{j=0}^{w-1} I(i,j) \quad (2.2)$$

If V(K) > global threshold the block is foreground region.

If V(K) < global threshold the block is background region.

In this case the global threshold value is (0.1) and the block size is (16).

b. **Normalization:** is the process used to standardize the intensity of an image by tuning the-grey-levels values according to desired range of values as [5]:

$$N(i,j) = \begin{cases} M_0 + \dfrac{\sqrt{V_0(I(i,j)-M)^2}}{\mathrm{V}} & if\ I(i,j) > M, \\ M_0 - \dfrac{\sqrt{V_0(I(i,j)-M)^2}}{\mathrm{V}} & else, \end{cases} \quad (2.3)$$

Where $\mathrm{I}(i,j)$ grey-level at pixel(i,j), $\mathrm{N}(i,j)$ is normalized grey-level at pixel(i,j), M and V are estimated mean and variance of (i,j), respectively. M_0 and V_0 are desired mean and variance, respectively.

c. **Orientation estimation:** is the process used to find the orientation (angles) of FP ridges, Figure 2-9 illustrates FP ridges orientations. The orientation calculation process is the first step or part of GF stages.

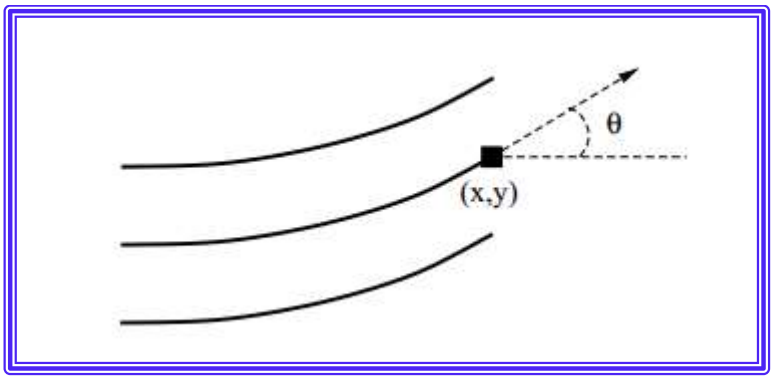

Figure 2-9: FP ridge orientations [6]

An algorithm proposed by Hong has been used for local ridges orientation calculation [5]. The algorithm steps are [6]:

i. Divide the normalized FP image into W × W block size centred at pixel (i,j) .

ii. Compute the gradient for each pixel in the block, $\partial_x(i,j)$ and $\partial_y(i,j)$ represent the gradient magnitude in x and y directions, respectively. Sobel operator has been used for gradient calculation. For computing the horizontal gradient $\partial_x(i,j)$ is done by convolving the normalized image with horizontal Sobel operator that defines as follows [6]:

$$\begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \tag{2.4}$$

For computing the vertical gradient ($\partial_y(i,j)$) is done by convolving the normalized image with vertical Sobel operator that defines as follows [6]:

$$\begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \tag{2.5}$$

iii. Using the following equations to estimate the local orientation at pixel(i,j) [6]:

$$V_x(i,j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} 2\partial_x(u,v)\partial_y(u,v) \tag{2.6}$$

$$V_y(i,j) = \sum_{u=i-\frac{w}{2}}^{i+\frac{w}{2}} \sum_{v=j-\frac{w}{2}}^{j+\frac{w}{2}} \partial_x^2(u,v)\partial_y^2(u,v) \tag{2.7}$$

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_y(i,j)}{V_x(i,j)} \quad (2.8)$$

Where $\theta(i,j)$ is the least square estimate of local orientation at the block with size of W × W centred at pixel(i,j).

iv. Using Gaussian filter to smooth the orientation in a local neighborhood, this done by converting the orientation image to continues vector field as follows [6]:

$$\Phi_x(i,j) = \cos\big(2\theta(i,j)\big) \quad (2.9)$$

$$\Phi_y(i,j) = \sin(2\theta(i,j)) \quad (2.10)$$

Where Φ_x and Φ_y are the component of vector field in x and y, respectively. After computing of vector filed now we apply Gaussian filter as follows [6].

$$\Phi'_x(i,j) = \sum_{u=-\frac{w\Phi}{2}}^{\frac{w\Phi}{2}} \sum_{v=-\frac{w\Phi}{2}}^{\frac{w\Phi}{2}} G(u,v)\Phi_x(i-uw, j-vw) \quad (2.11)$$

$$\Phi'_y(i,j) = \sum_{u=-\frac{w\Phi}{2}}^{\frac{w\Phi}{2}} \sum_{v=-\frac{w\Phi}{2}}^{\frac{w\Phi}{2}} G(u,v)\Phi_y(i-uw, j-vw) \quad (2.12)$$

Where G is a Gaussian low-pass filter with size of wΦ × wΦ. After smoothing has been calculated, the final orientation is (O) and defines as follows [6]:

$$O(i,j) = \frac{1}{2}\tan^{-1}\frac{\Phi'_x(i,j)}{\Phi'_y(i,j)} \quad (2.13)$$

d. Ridge frequency estimation: is the process of estimating the local ridges frequency, ridge frequency is another parameter to implement GF. The ridge frequency depicted in Figure 2-10 (a).

The ridges frequency estimation process has been achieved by using algorithm proposed by Raymond Thai, the algorithm steps are [6]:

i. Divide the image into blocks for each block size is W × W.

ii. Project the grey-levels values of all pixels inside each block, the projection done with the direction orthogonal to the local ridge orientation. The sinusoidal-shape wave almost represents this projection as shown in Figure 2-10 (b).

iii. Computing the ridge spacing $S(i,j)$ by counting the number of pixels between successive minimum points in the projections waveforms.

iv. Finally, computing the ridges frequency for a block centred at pixel (i,j) as follows:

$$F(i,j) = \frac{1}{S(i,j)} \quad (2.14)$$

Where $F(i,j)$ is the local ridge frequency.

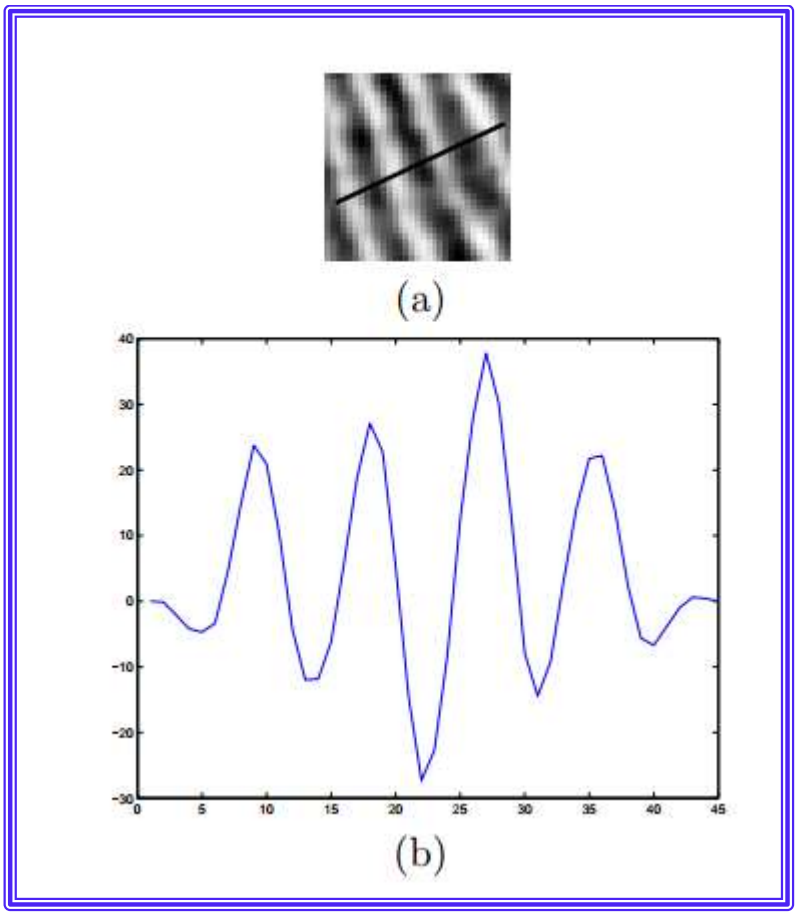

Figure 2-10: (a) 32x32 FP image block represent ridge frequency and (b) wave form projection of the block in (a) [6].

e. Gabor filtering: after computing the ridge orientations and frequency, these parameters are used to construct the GF. An even-symmetric two dimensional GF is used to repair the ridges structure and noise decreasing. Fig. 2-11 illustrates GF envelope in the spatial domain. The even-symmetric GF is real part of Gabor function, which is represented by cosine wave modulated by the Gaussian envelope.

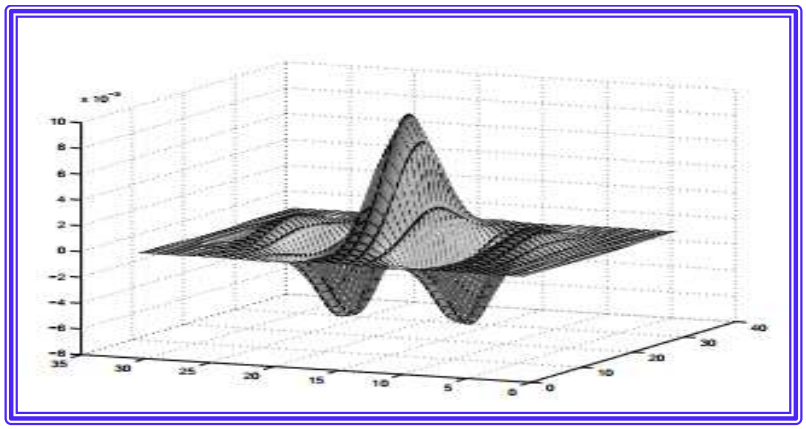

Figure 2-11: an even-symmetric GF [6]

The definition of GF in spatial domain is given as follows [6]:

$$G(x, y, \theta, f) = e^{\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2}+\frac{y_\theta^2}{\sigma_y^2}\right]\right\}} \cos(2\pi f x_0) \tag{2.15}$$

$$x_0 = x\cos\theta + y\sin\theta \tag{2.16}$$

$$y_0 = y\cos\theta - x\sin\theta \tag{2.17}$$

Where θ is the orientation of the GF, f is the frequency of the cosine wave, σ_x and σy are the standard deviations of the Gaussian envelope along the x and y axes, respectively, and x_θ and y_θ define the x and y axes of the filter coordinate frame, respectively.

At this point the GF can be applied to the FP image, this done by convolving the normalized FP image with the transfer function of GF in the spatial domain as follows [6].

$$E(i,j) = \sum_{u=-\frac{wx}{2}}^{\frac{wx}{2}} \sum_{v=-\frac{wy}{2}}^{\frac{wy}{2}} G\big(u, v; O(i,j), F(i,j)\big) N(i-u, j-v) \tag{2.18}$$

Where (O) is the orientation image, F is the ridge frequency image, N is the normalized FP image, and wx and wy are the width and height of the GF mask, respectively.

2.6.3 Features Extraction

Mainly AFIS uses the local features of FP image which is called minutiae (edge-termination and bifurcation) to make recognition among all FPs as well as there are another types of feature can be used in FP classification and identification like statistical features and texture features.

A critical step in AFIS is reliably extracting minutiae from the input FPs images, this step generally consist of the following sub-steps [40]:

1. Binarization: is the process of converting a gray FP image to binary image by using globule thresholding. If a pixel value equal or greater than threshold value it become “1”, otherwise it become “0”.
2. Thinning: is the process of making the ridges width with one pixel to generate skeleton image. This process is used to simplify minutiae detection process.
3. Extract the minutiae from the skeleton image by using an appropriate algorithm such as Crossing Number (CN) algorithm, which will be discussed late on
4. Post-processing the minutiae set according to some heuristic rules and the duality property

The most famous algorithm for minutiae extraction is Crossing Number (CN). CN algorithm is used in this research for minutiae features extraction, CN algorithm can be implemented in following sub-steps [14]:

a. Segment the thinned FP image into 3x3 blocks size as shown in the Fig. 2-12.

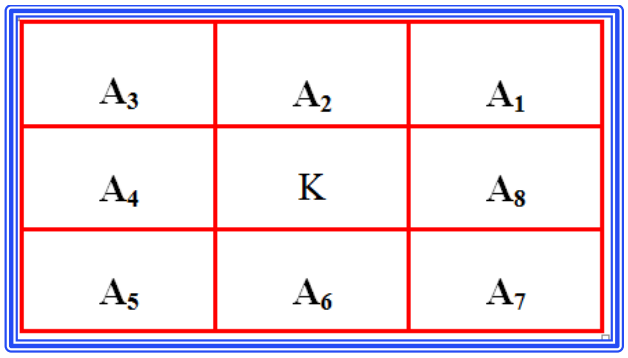

Figure 2-12: Checkpoint and neighboring pixels [14]

Where K is the checkpoint pixel, A_1, A2… A_8 are the neighboring pixels which are arranged in anti-clockwise direction.

b. Checking the K type (ridge ending or bifurcation), this done by calculation the CN of the pixels around K point as follows [14]:

$$CN = \sum_{i=1}^{8} |A_{i+1} - A_i|, \qquad where \quad A_9 = A_1 \tag{2.19}$$

If CN is equal to (1) the ridge type is (ridge ending) if CN is equal to (3) ridge type is (bifurcation). Figure 2-13 shows the detected minutiae format in the neighboring block.

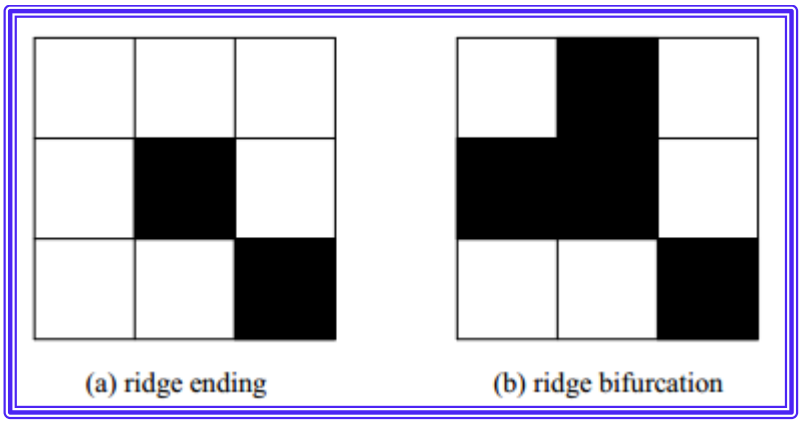

Figure 2-13: the detected minutiae ridges (ending and bifurcation) [14]

2.6.4 FP Matching

The FP matching is the process of comparing two given FPs and returns the matching result or degree of similarity (which be either a score between 0 and 100 or binary decision matched/non-matched) [27]. The FP matching can be classified into three families [41]:

1. **Correlation-based matching:** in this approach two FP images are composed and the correlation between identical pixels is computed for different alignment (e.g. Various displacements and rotation).
2. **Minutiae-based matching:** This is the most popular and widely used technique, being the basis of the FP comparison made by FP examiners. Minutiae are extracted from the two FPs and stored as sets of points in the two-dimensional plane. Minutiae based matching essentially consists of finding the alignment between the template and the input minutiae sets that results in the maximum number of minutiae pairings.

3. **Non-Minutiae feature-based matching:** minutiae extraction is difficult in extremely low-quality FP images. While some other features of the FP ridge pattern (e.g., local orientation and frequency, ridge shape, texture information) may be extracted more reliably than minutiae, their distinctiveness as well as persistence is generally lower. The approaches belonging to this family compare FPs in term of features extracted from the ridge pattern.

2.7 Identity Management in E-Government Applications

Along with the increase of the number of web services, users of such services were increasing rapidly, more credentials and identities were issued and that made the management of them became more challenging for both of users and service providers. Identity Management was presented to ease the online management process of user identities that resulted in different identity management systems. These systems are not interoperable, that means the identity authentication implemented in a specific system wasn't recognized by the others. This problem was solved by introducing Identity Federation [42].

Identity management is an automated, centralized approach used for providing enterprise wide access to the resources for authorized individuals and employees. Identity management is focusing on defining an identity for each user (process or human), attributes association with the identity and providing methods for the user to verify the identity. The main concept of the identity management system is the use of Single Sign On (SSO). SSO used to help the user to access all network resources with single authentication [43].

Federated Identity Management (FIM) indicates the standards, agreements, and the technologies that enable identities portability, identity attributes and privileges across different multiple enterprises and several applications that support thousands and even millions of users. The implementation of the interoperable federated identity schemes in multiple organizations gives the ability of SSO for each employee in any organization to access services within the federation based on the trusted relationships that associates with the identity. The primary function of federated identity management is the identity mapping. Different security domains represented differently by identities and attributes [43]. Figure 2-14 shows FIM operations.

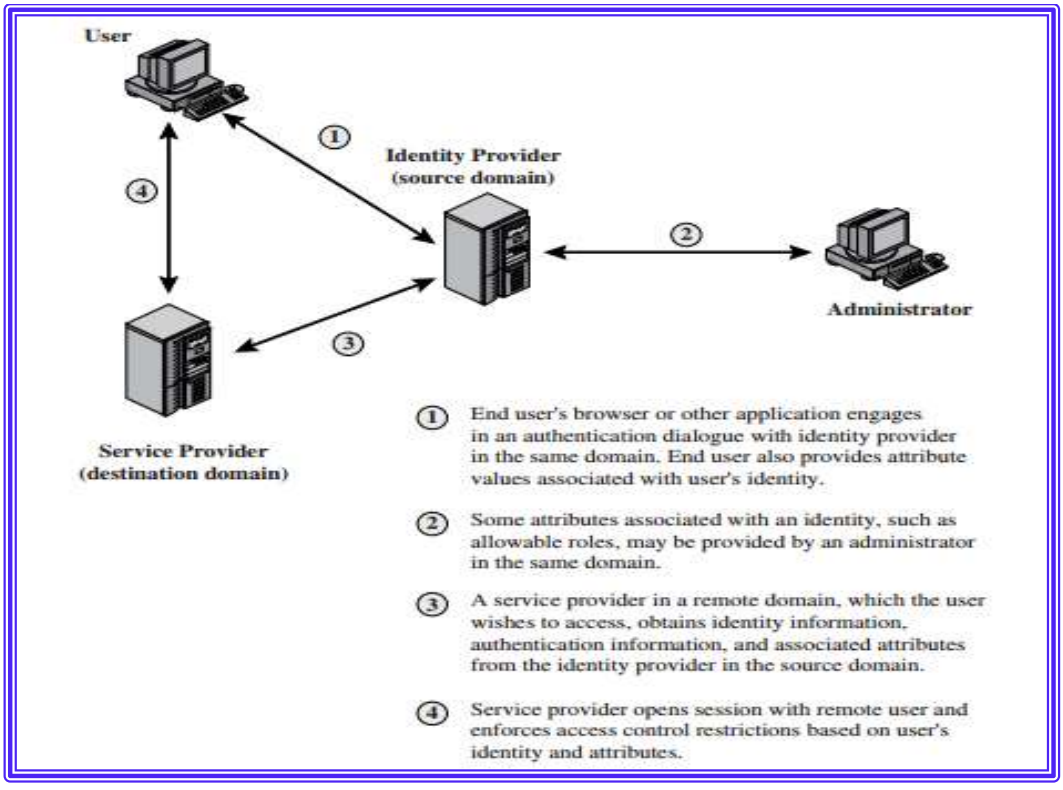

Figure 2-14: FIM operation [43]

2.8 National Identification System of Indian Residents

This system is to provide a unique ID for each and every Indian citizens. Each citizen of a country can have unique 16 digits Unique Identification Number (UID) by using unique identification system. Identification number has important application in our life for example it can helps the governments to track down individuals, public-private service, and other E-Government transactions [44].

2.8.1 National Unique Identification System Phases

a. **Recording phase**: during this phase the system takes the personal information (morphological information) such as name, sex, mobile number date of birth, etc., and store this information in central Database. Generate of 16 digit unique number for each citizen as ID number, this number created randomly by main computer. The biometric information such as iris and ten fingerprints can be used in verification process and reduce the de-duplication which produced from similarity in registration information [44].
b. **Authentication phase**: after the user complete his/he registration, he is ready to try his identity and login to the system by introduce his ID number, the system will back the information associated (password or biometric information) with this ID and checks with what given by the user, finally the system return the result of matching in Yes for correct matching or No for false matching [44].

2.8.2 National Unique Identification System Advantages and Disadvantages

Advantages:

1. Single unique number is used to decreasing all manual work. This increasing efficiency for every details and reducing database maintenance efforts.
2. Security and verification will be increased by using Biometric Authentication technology,
3. Electricity bill and telephone bill, book railway ticket and airline ticket and many other public services will be provided by the system.

Disadvantages:

1. The system need network to be very quick and crash free, which is very expensive.
2. Robust security measures are needed in this system.

The National Unique Identification System is managed by the Unique Identification Management (UIDM), which is the core of the system and play an important role in the administration process of all system transactions. Figure 2-15 shows the data flow in UIDM [45].

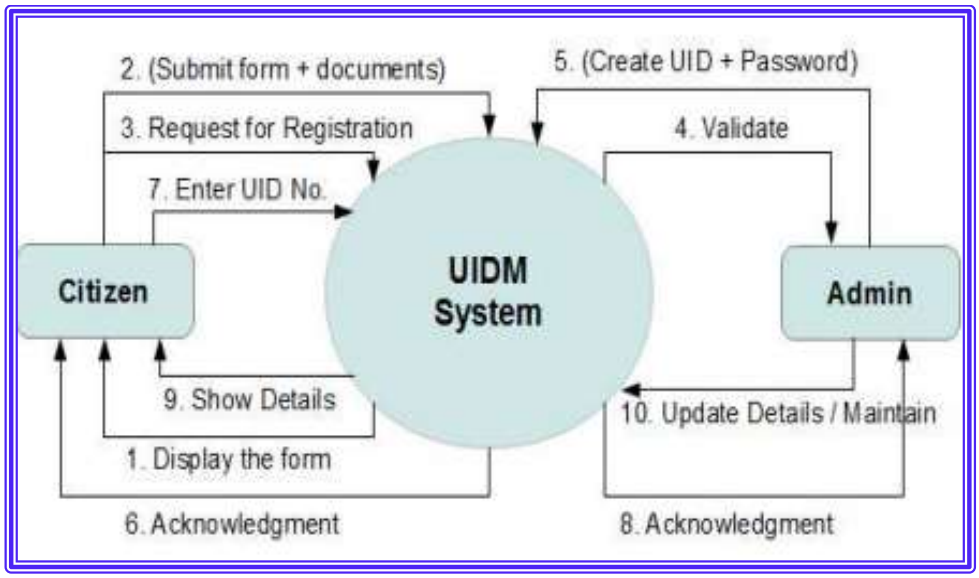

Figure 2-15: UIDM data flow [45]

2.9 Quick Response Code (QR)

A QR is a matrix code, invented by Toyota Company in Japan in 1994. The purpose of QR was to assist track auto parts during production. QR is a two dimensional (2D) barcode matrix which saves data in two dimensions form. Data is performed as square dots with constant pattern Includes the two dimensions (horizontal and vertical). Special imaging devices is used to capture QR image and retrieved its included contains called (QR scanners). 2D barcode contains more amounts of data than one dimensional (1D) barcode. From security view 2D barcode is more secure than 1D barcode [46, 47].

QR modules execute number of functions, some of modules perform row data storage, while other modules perform some of control function such as improves the performance of QR image capturing, error repeating and distortion

restoration. Timing pattern allows the QR scanners know the symbol size. There is a wanted “quite zone”, no data included in a four-modules wide buffer region, but to guarantee that encirclement rounding text or marking are not wrong for QR code data. In the three corners of QR symbol, special position-detection pattern are located to solve the problem of detected symbol orientations and position (x,y). The design strategy of these patterns allow to detect the symbol from any direction within a full 360 degrees Fig. 2-16 reveals QR modules [48]

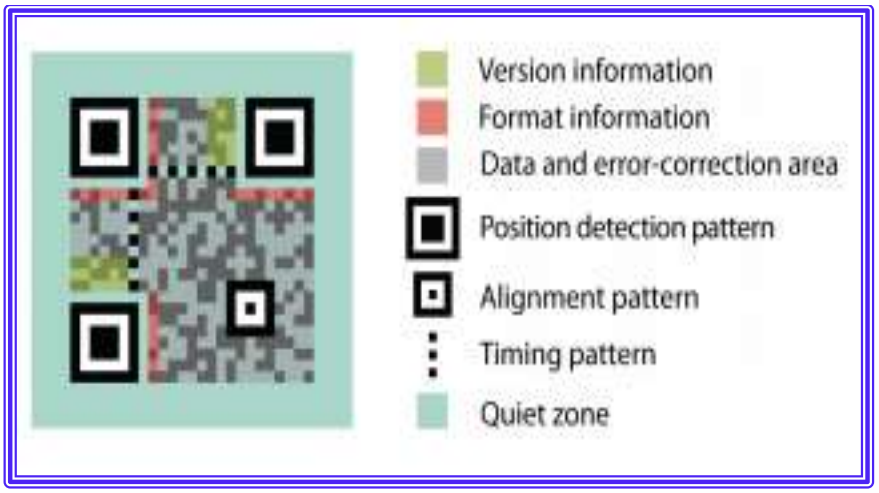

Figure 2-16: QR structure

2.10 Hash Function

Hash function is a one way cryptographic or encryption function, the input of hash function is variable-length block of data and output is a fixed-length hash value. The main purpose of hashing is data integrity. When a message authentication is provided by hash function, the output of hash function (hash) value is called message digest. The hash code is produced by applying the hash function to all bits of the input message, any change act in bit or bits of input

message will cause changes in hash code. The hash value (h) is generated by function (H) as in the form [43].

$$f = H(M) \tag{2-20}$$

Where M is variable-length input message and H (M) is fixed-length hash value. Fig. 2-16 illustrates the structural generation of secure hash code.

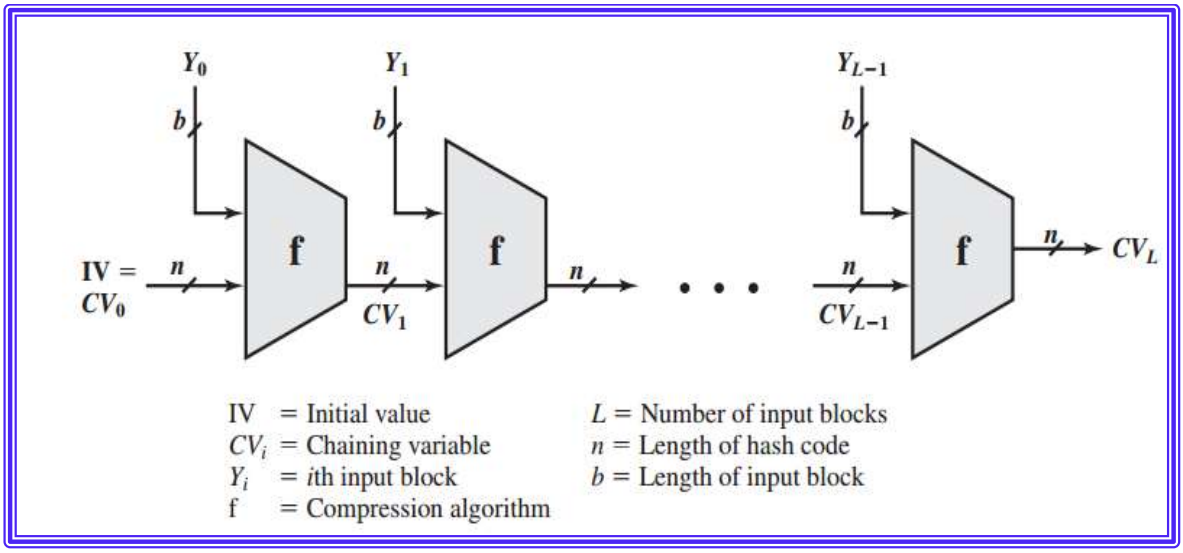

Figure 2-17: general structure of generation secure hash code [43]

Hash function involves cascaded stages of compression function (f) that take two inputs (an n-bit input from previous step, called chaining variable, and a b-bit block) and generate an n-bit output. The chaining variable has initial value at the start of hashing which is specified by the procedure of algorithm. The last value of chaining variable is hash value. The hash function can be summarized as follow:

$$Cv_o = Iv = \text{initial n-bit value} \tag{2.21}$$

$Cv_i = f(Cv_{i-1}, Y_{i-1})$ for $1 \leq i \leq L$, (2.22)

$H(M) = Cv_L$ (2-23)

Where the input of the hash function is a message M consisting of blocks $Y_o, Y_1, ..., Y_{L1}$.

2.11 Message Digest Five (MD5)

MD5 is a one way hashing function with variable length input data and fixed length 128-bit output hashing value. It is the development of MD4 which is completely faster than MD5 because MD4 produce hash value by three rounds while MD5 contains four rounds. The main steps of MD5 algorithm are [49]:

a. **Padding bits behind the input message:** padding of bits is imposed, "0" and "1" bits are added behind the input message to make its length congruent to 448mod512. First, a single bit "1" is appended to the message. Then, a series of "0" bits are appended so that length (the padded message) ≡ 448 mod 512.
b. **Add 64-bit binary string which is the representation of the message's length:** this means of adding the original message's length in format of 64-bit in hexadecimal form to the padded message's tail. For example if a message with length of 1000 bits, the hexadecimal of number 1000 which is 0x00000000000003E8 will be added.

a. **Divide the input into 512-bit blocks:** in this step the padded message will segmented into 512-bit blocks like m1, m2…mn.

b. Initialize chaining variables: a buffer (4 registers) is used to hold the intermediate and final result of hash function that are each 32 bits long known as chaining variables. The buffers (A, B, C, and D) are defined as:

A= 01 23 45 67

B= 89 ab cd ef

C= fe dc ba 98

D= 76 54 32 10

c. Process Blocks: the input message is joined with the four chaining (A, B, C and D) using four auxiliary functions (F, G, H and I). There are four rounds and each round consists of 16 essential operations. By using message word M_i and fixed K_i, processing function (F) is applied to the 4 buffers (A, B, C and D) as shown in figure 2-18. The term "<<<s" is represents a left shift for binary numbers by s bits. The four auxiliary function apply some of logical operations such as AND (∧), OR (∨), NOT (-) and XOR (⊕) to the input of three 32-bit and produce output of 32-bit. The following terms represents the auxiliary function.

F (B, C, D) = (B∧C) ∨ (-B∧D)

G (B, C, D) = (B∧C) ∨ (C ∧ -D)

H (B, C, D) = B ⊕ C ⊕ D

I (B, C, D) = C ⊕ (B ∨ -D)

The functions G, H and I are similar to the function F, in that they act in "bitwise parallel" to produce their output from the bits of B, C, and D

d. Hashed output: the output value is 128-bit generated after 4 round of identical iteration.

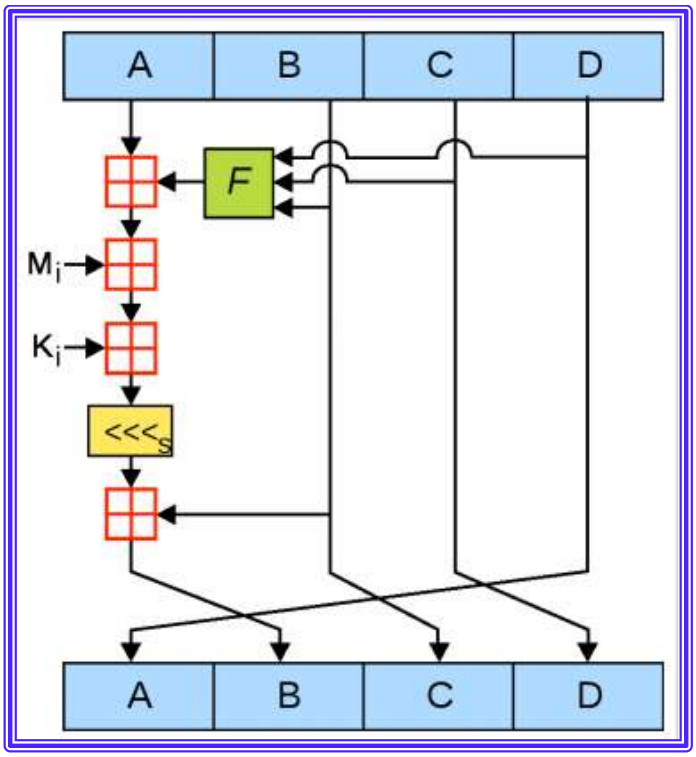

Figure 2-18: MD5 iteration

2.12 Secure Socket Layer protocol (SSL)

SSL is a cryptographic protocol has a wide range of application that provides secure connection to web servers. SSL depends on cryptographic functions in connection security. SSL has been accepted as security protocol in some of standard version that released by Netscape and Internet Explorer. SSL is located between Transmission Control Protocol (TCP) and an application protocols such as Hyper Text Transfer Protocol (HTTP). In the presence of SSL HTTP is upgraded to be secure HTTP (HTTPS) that provide security requirement for web based application [50]. Figure 2-19 illustrates SSL position and its protocols.

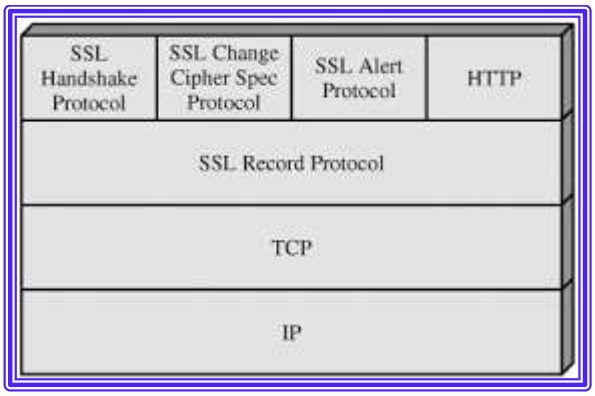

Figure 2-19: SSL protocol stack [43]

2.12.1 SSL functions

1. **Authentication**: SSL provides authentication function by allowing the client to identify the server, server identifies the client is optional. Public key with digital certificate is the way that has been used by SSL in authentication providing.
2. **Confidentiality:** the second security function that provided by SSL is encrypted data transmission is used to protect the transmitted data and ensures data reading by authorized users only. A symmetric encryption approach has been used to perform this activity.
3. **Integrity:** in which the transmitted data arrive without any modification by an unwanted parity during sending time. Different approaches such as message digest and checksum value are used for integrity providing.

2.12.2 SSL Protocols

There are three main protocols in SSL [51]:

a. **Handshake protocol:** client and server authentication is achieved during this phase, Handshake protocol starts in the occurrence of communication between a client and a server. The activities that done by Handshake protocol are; certificate exchange, key exchange and identity authentication as shown in figure 2-20.

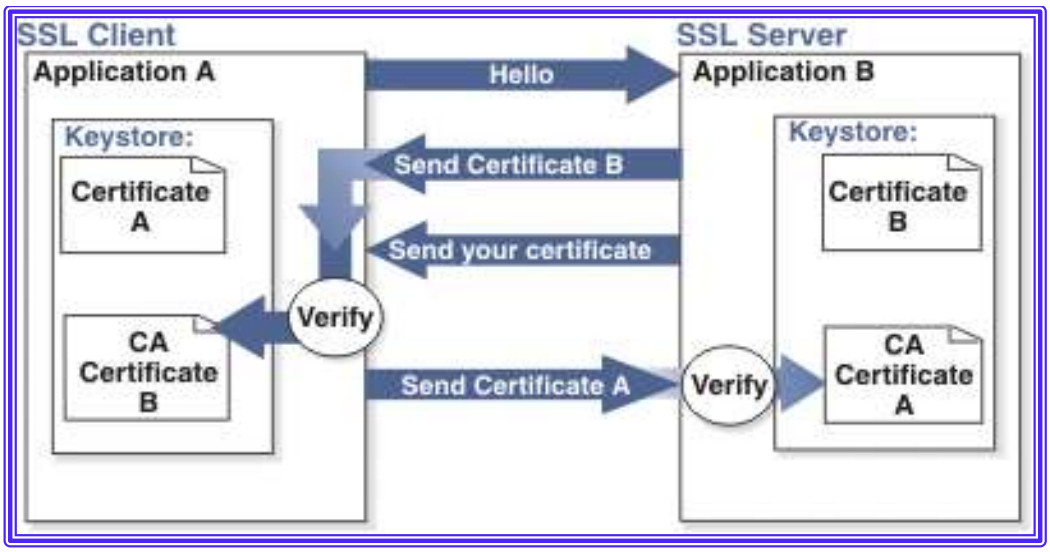

Figure 2-20: SSL configuration

Different steps are represented in handshake protocol:

1. A client sends a Hello message to the server asking for a SSL connection. All parameters that related to the SSL version support, random number, cipher suits and compression method are included in this message.

2. Server Hello message will respond to the Client Hello message. This contains the selected protocol version, random number, cipher suits and compression method. Session ID may be send in this phase.
3. The server sends its Certificate to the client for authentication purpose.
4. The server sends alert message (Hello Done Message) to indicate the client on finishing the handshake phase.
5. The client replies by a Client Key Exchange message that contains pre-master key, public key or nothings.
6. The client and server compute the common master secret key from the proposed two keys (pre-maser and random number). The maser secret is used to compute a key-session, the other key data for this connection are created from key-session. Key data includes; client-write key and a server-write key.
7. The client sends a Change Cipher Spec record to tell the server "Everything I tell you from now on will be encrypted".
8. Client's finish message has been sent which is encrypted by the client-write key
9. The server will decrypt the client's finished message by the server-write key. Another activity has been achieved in this phase which is hash and MAC verification. If verification failed, handshake will be failed and connection will be dropped.
10. Is the last step, server's finished message and Cipher Change Spec have been sent. Also, the client decrypts and verifies the received server's message.

After these 10 phases the “handshake” is finished and application messages are ready exchange between the two sides (client-server).

b. **Record protocol:** the next step after handshake completion is the Record protocol starting. Record protocol is used to provide confidentiality and integrity for application data by encapsulates application data using the session key that produced during Handshake protocol.
c. **Alert protocol:** Is responsible on alerting an error that appears during the time of application data exchange.

Chapter Three

Proposed System Design

3.1 Introduction

This chapter includes design details of the Secure Network Authentication Based on Fingerprint National Identification Number. Section 3.2 describes the system design layout. Section 3.3 describes the system architecture. Section 3.3 describes the algorithm of the National Identification Number (NIDN) generation which includes fingerprint image processing, combining image features with user credentials and generation of NIDN using hashing function. Other system service design and it's flowcharts also described.

3.2 System Overview

The proposed system is a client-server interaction, connected either locally using LAN or via Internet by public IP. Figure 3-1 illustrates the proposed system layout. The proposed system provides an authentication operation for different types of applications such as E-Government applications (normal or sensitive applications). Normal application is simple authentication service provided to the users to access applications such as registration, national services …etc. Sensitive application includes application that deals with sensitive information such as elections, banking transaction, passport application …etc. The client side provides front-end interface to the users for registration process and authentication process as well as web camera and FP scanner connected in the client side for capturing QR and FP images respectively. The server side

provides the processing functions for system's operations such as FP image enhancement, features extraction, DB registration and user' verification.

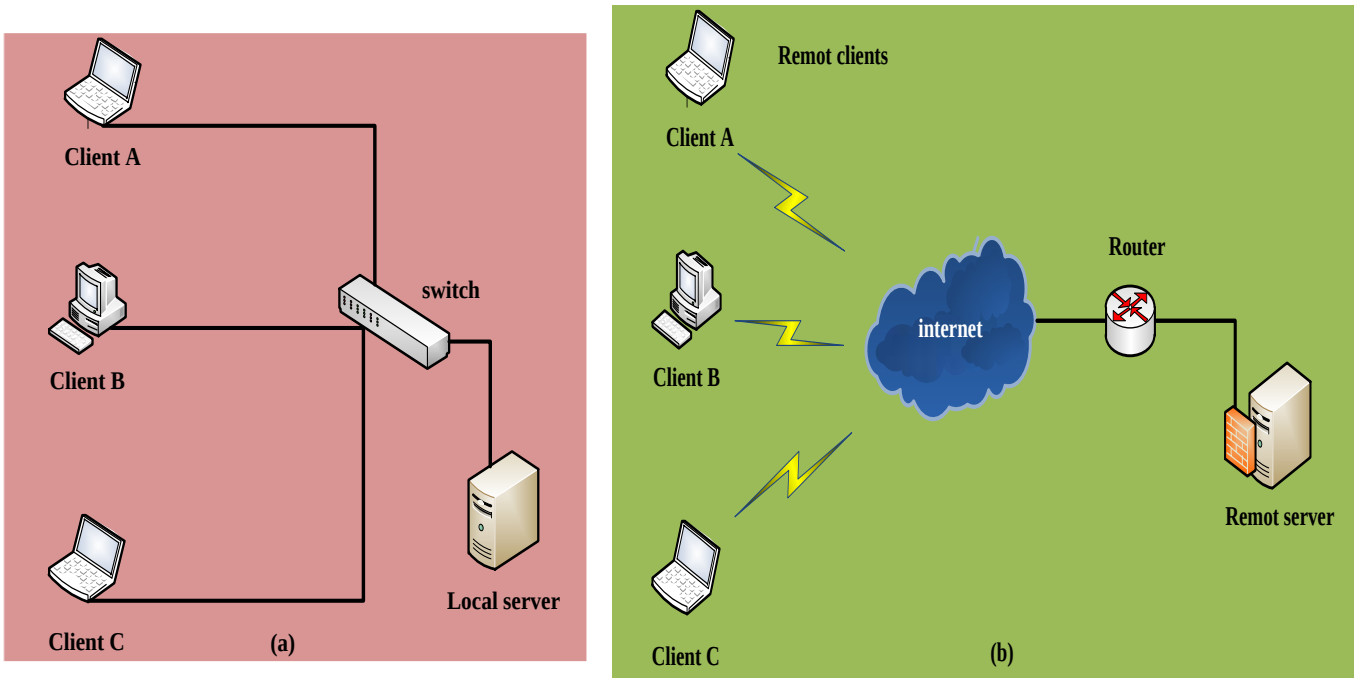

Figure 3-1: System layouts; (a) local connection and (b) via internet connection

3.3 System Architecture

System architecture gives further view than system layout by defining system's modules and sub-modules for each side (client and server). Figure 3-2 illustrates the system architecture.

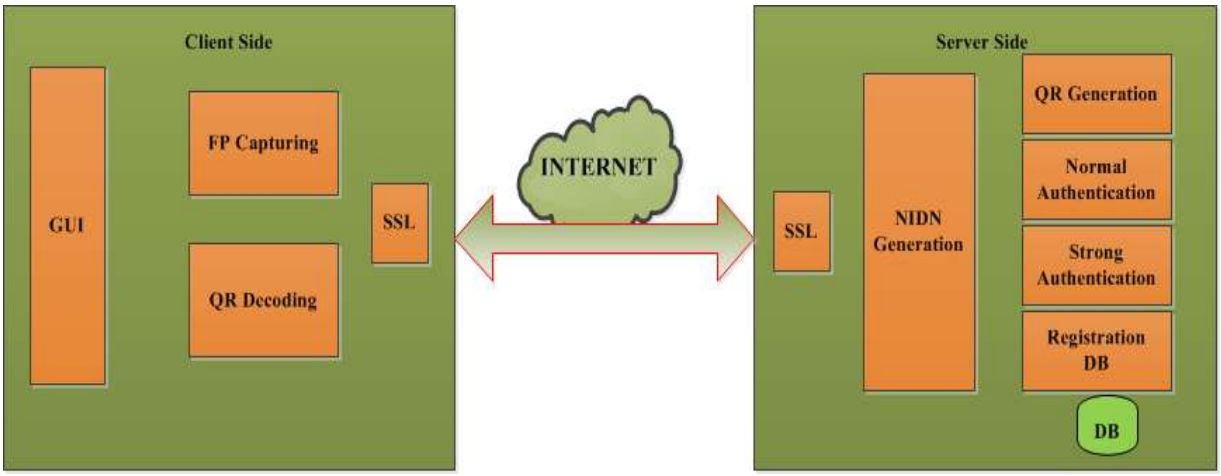

Figure 3-2: System architecture

3.4 NIDN Generation

The main purpose of this research is to generate NIDN based on combination between fingerprint minutiae features and some personal information (birthdate and name). This number will be included inside QR code as secure machine readable number and instead of using expensive chip memory card. The main stages of NIDN generation are pre-processing stage, minutiae extraction stage, combing minutiae with personal information stage and hashing stage using MD5 hashing function. Figure 3-3 illustrate NIDN flowchart.

The NIDN generation algorithm can be abstracted as follow:

Goal: Generate Unique National Identification Number.

Input: FP image + personal information (name + birthdate).

Output: Fixed length 128-bit NIDN.

Step1: FP image pre-processing to enhance the quality of the FP image and make the minutiae features extraction more reliable using the FP enhancement algorithm based on GF described in the previous chapter section (2.6.2).

Step2: Minutiae extraction using CN algorithm described in previous chapter section (2.6.3) and matrix features generation.

Step3: Combine the minutiae features with the credential personal information as one block of combination between these two types of information.

Step4: Enter the combination block to one-way MD5 hashing function to compress the input block into fixed-length output of 128-bit as NIDN.

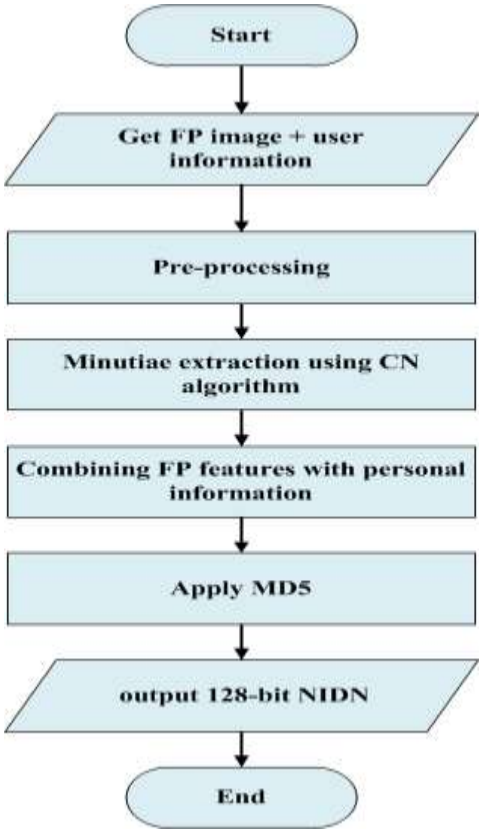

Figure 3-3: NIDN generation flowchart

3.4.1 Pre-Processing

Image pre-processing is the first step in NIDN generation. This step includes; FP image enhancement, binarization and thinning as illustrated in Figure 3-4.

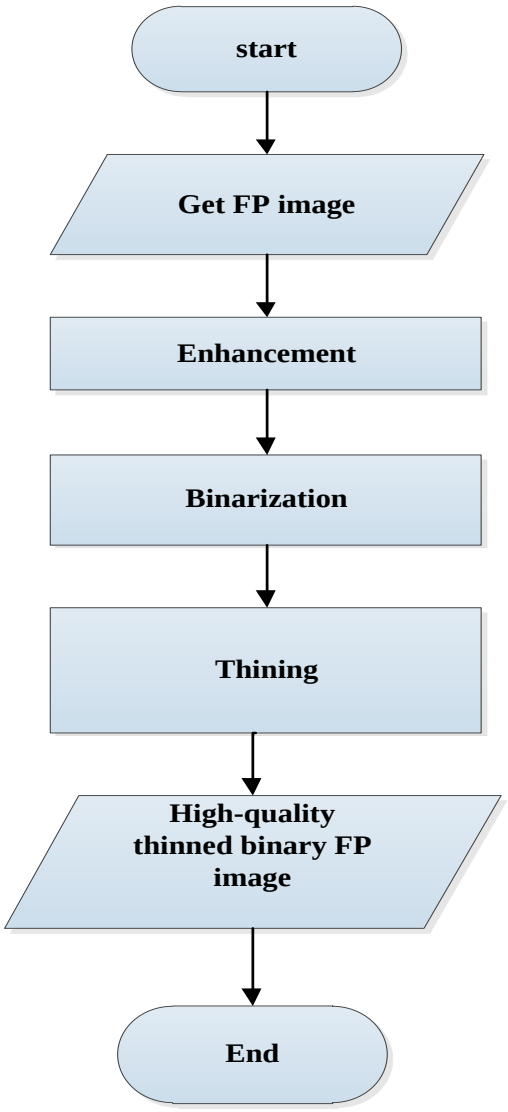

Figure 3-4: Pre-processing flowchart

a. **Enhancement:** in this process the FP quality is enhanced by removing noises and sharpening ridges. This sub-stage achieved by using the proposed FP image enhancement algorithm which is described in chapter two, section 2.6.2 (b).
b. **Binarization**: in this process the enhanced FP image converted from grey-level into binary-level "0" and "1", this done by using the thresholding decision for each pixel value in the grey-levels.
c. **Thinning:** in this process the binary FP image is thinned to generate the skeleton image by making the ridges width one pixel, this is done by using morphological operation "thinning".

3.4.2 Minutiae Extraction

After FP pre-processing stage the minutiae features extraction stage starts. The purpose of this stage is to extract FP minutiae that represent by ridge's type, position and orientation. These features are represented in form of matrix called "minutiae matrix".

The minutiae detection is based on the CN algorithm which is described in the chapter two, section 2.6.3. The minutiae matrix consists of three columns; position, type and orientation and variable number of rows according to number of minutiae located in the FP image. Each matrix column can be described as follow:

a. **Position:** represent the location of the minutiae x and y according to the FP image coordination and this done by finding the detected pixel location(i, j).

b. Type: represent the minutiae type which is either edge termination or bifurcation, by using CN algorithm.

c. Orientation: represent the angle of the detected minutiae such as the orientation of the FP ridge as described in chapter two. This is calculated by finding the orientation of a certain minutiae from the orientation image which calculated during ridge estimation phase. Figure 3-5 illustrates the minutiae features matrix.

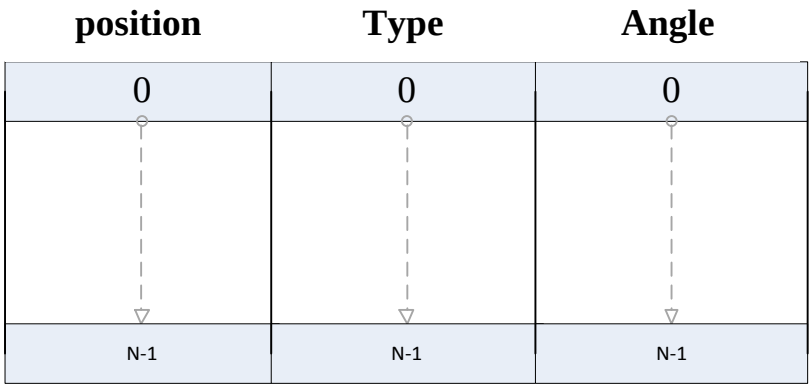

Figure 3-5: Minutiae features matrix

3.4.3 Combining

In this stage the personal information (name and birth date) will be combined with FP minutiae features to avoid the similarity in fingerprint features if happened.

3.4.4 Applying MD5

In this stage the mixed block of combination (FP minutiae + personal features) feed into one-way hashing function (MD5) to generate fixed length 128-bit NIDN represented in 32-digits hexadecimal.

3.5 Registration

After the completion of the process of NIDN generation, the registration process starts to save the users information in the system DB. The information to be saved are; user NIDN, user name, user birthdate and user FP image of forefinger. The designs of the system DB consist of one table with four columns; U-NID, U-Name, U-Birthdate and U-FP image. The NIDN is the primary key of the system DB. Figure 3-6 reveals system DB.

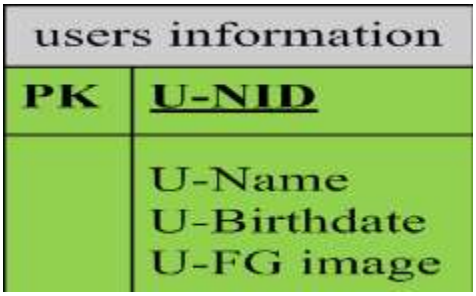

Figure 3-6: System DB

More information can be included in the DB as required by the application of the system, for example; user's birthplace, gender …etc. Figure 3-7 illustrates the registration flowchart.

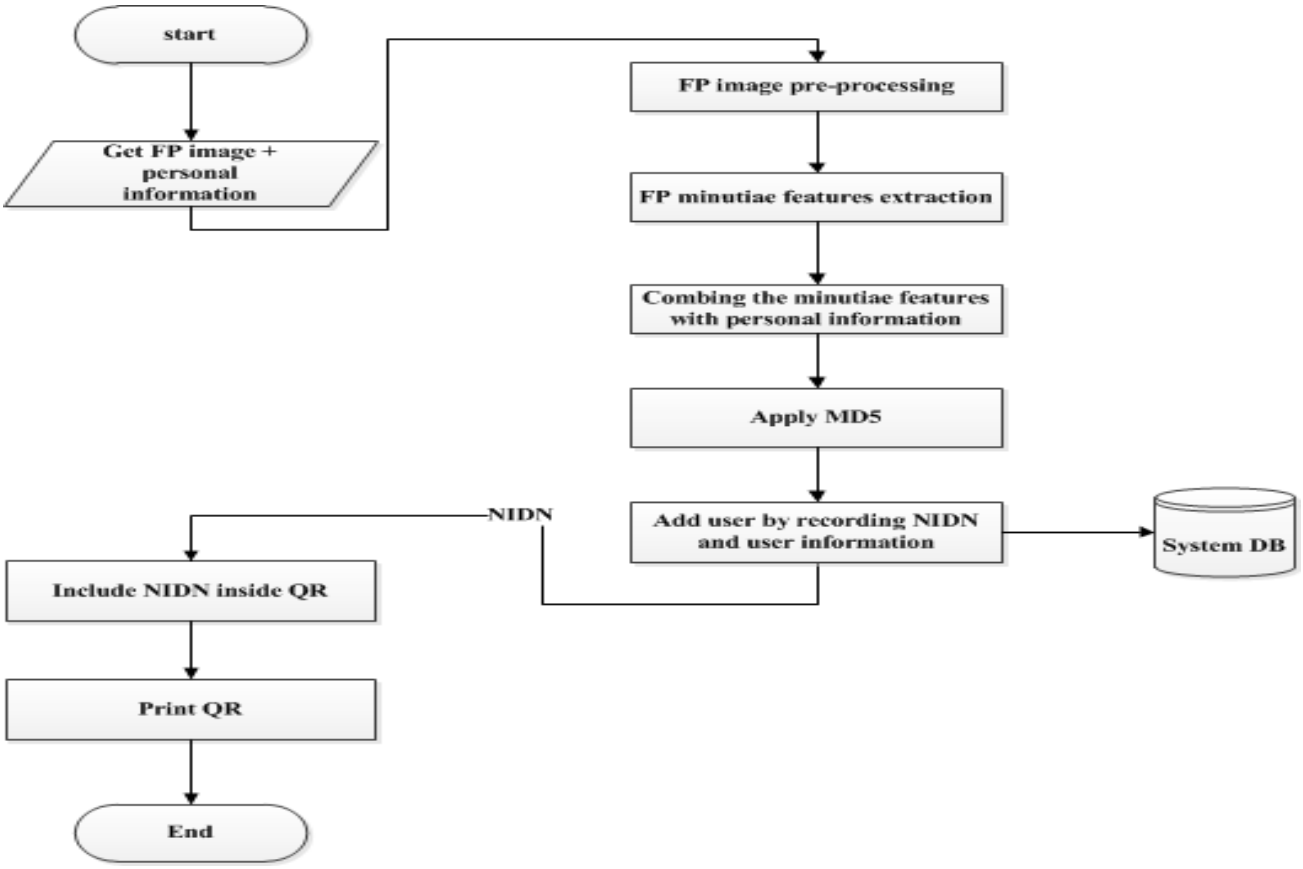

Figure 3-7: Registration flowchart

3.6 Normal Authentication

The normal authentication process is one of the authentication services provided to support the normal applications authenticity. The normal application is the applications that does not need strong authentication because it does not contains sensitive information. In this process the system will capture NIDN from user QR card for identification and verification, the system compare the input NIDN with all stored NIDN in the system DB, if the comparison is found, the user is accepted for public services, otherwise the user denied from public services. Figure 3-8 shows registration flowchart.

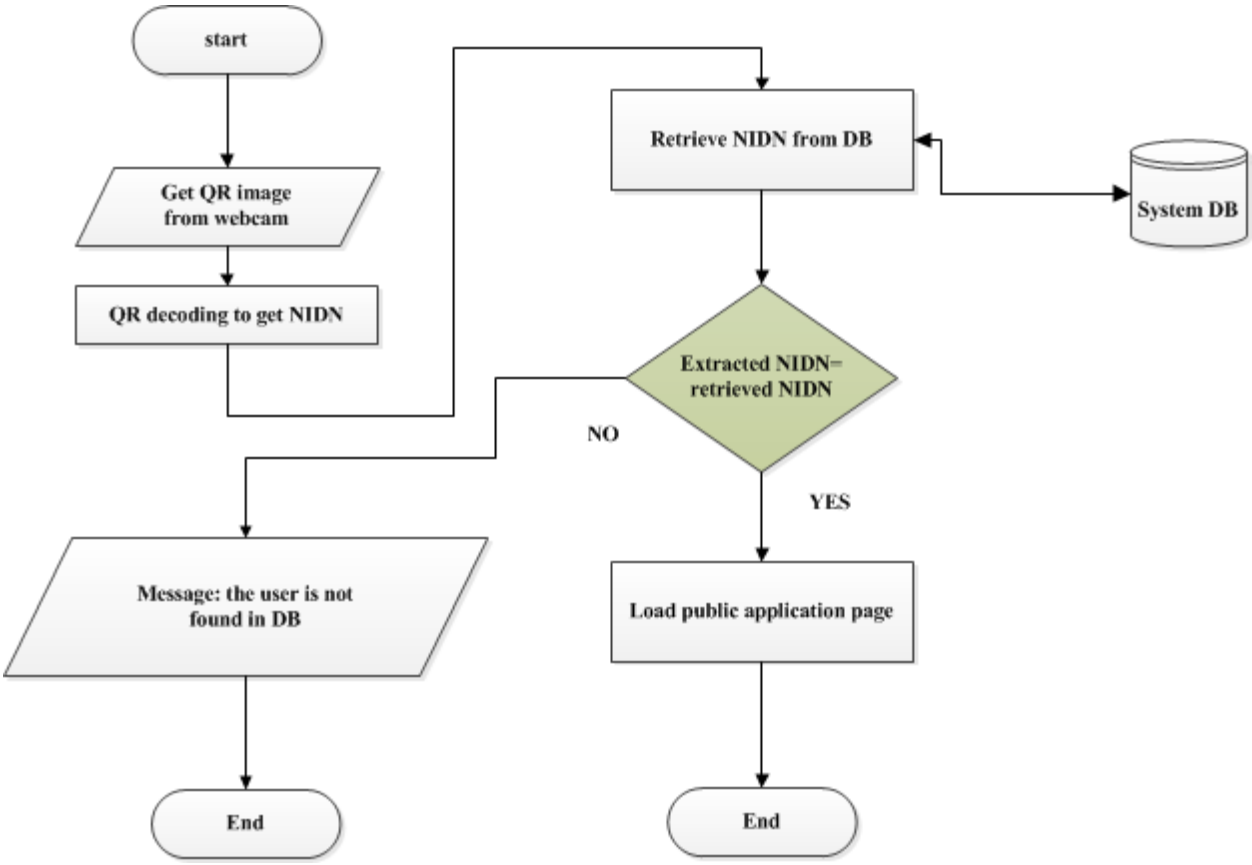

Figure 3-8: Normal authentication flowchart

3.7 Strong Authentication

The sensitive information such as banking information, credit card and ATM transaction, need strong authentication for user's identity. In the proposed system, strong authentication is done by using NIDN as user's identifier and FP of the forefinger of the right hand as user's verifier, this scheme verify the users physically by their FP image, figure 3-9 illustrates strong authentication process. Strong authentication takes two inputs; NIDN and FP image, the user presents his NIDN for the system as identifier, the system will check the user NIDN in the system DB as in the normal authentication process, two decisions are

generated from this process; first if user does not found in the system DB, in this case the user will be denied from the system's services, otherwise the system will retrieve the FP image of the user that is saved during registration phase in the system DB. At this situation two FP image are available for the same user, one-to-one verification between these FP images is applied. One-to-one verification will return the matching result between these FP images, this result is called "matching score". User access depends on the matching score value by comparing this value with predefined threshold value. Figure 3-10 illustrates strong authentication flowchart.

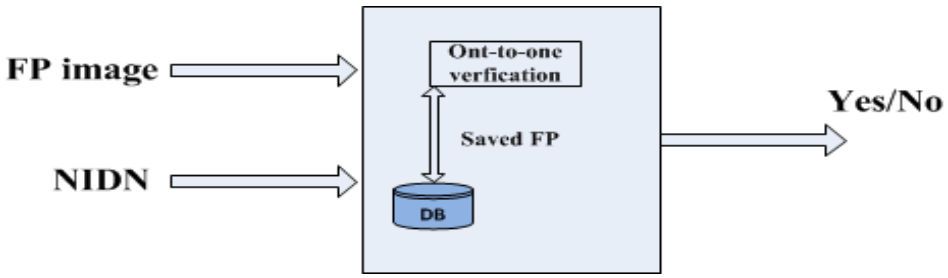

Figure 3-9: Strong authentication process

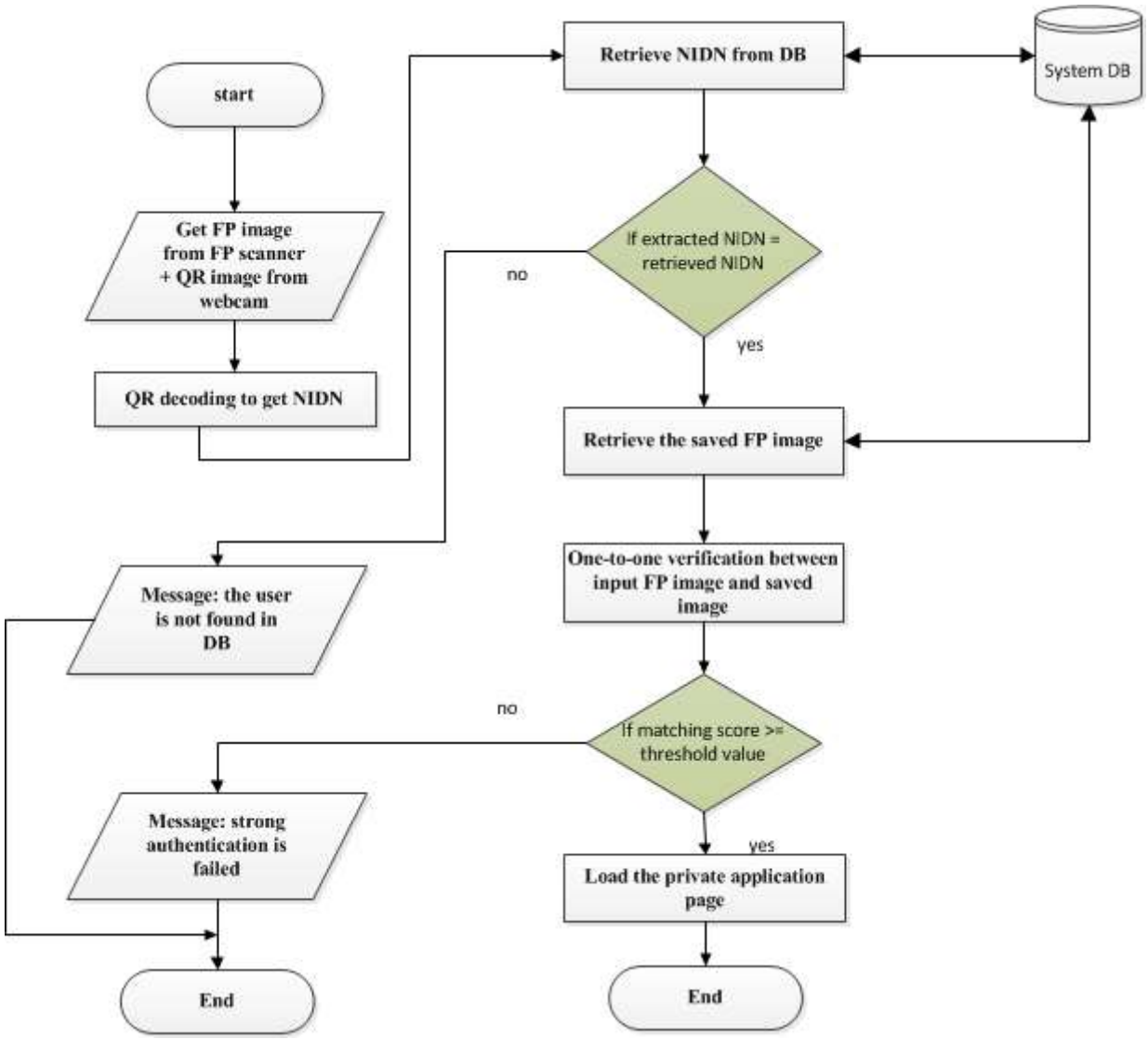

Figure 3-10: strong authentication flowchart.

For one-to-one FP verification Source AFIS verification algorithm has been used. Source AFIS is a multi-purpose FP recognition algorithm developed by Robert Važan, is a good FP matching algorithm and provides one-one verification [1]. The algorithm has been adopted in this research for the following reasons:

a. **Performance:** this algorithm provides high performance as follow:

1. Extraction time, 180 ms
2. Matching speed, 10000 FP/sec
3. Template size, 0.26 KB on average

b. Accuracy: this algorithm provided high level of accuracy as follow:

1. 3.6% Equal Error Rate (EER), 10.9% False Rejection Rate (FRR) and 0.01% False Acceptance Rate (FAR) in 1:1 verification

c. Software Development Kit (SDK): this engine provided SDK for different programming language such as C#.NET, Java, VB.NET and ASP.NET.

3.8 QR Coding

The QR coding is the process of generating QR for the NIDN, this process done in the server side by including NIDN that generated during NIDN generation phase inside QR image. The reasons of QR using are:

a. Machine readable: QR is readable by computer and quickly retrieves its content, in this case the NIDN can be captured for user's login via special scanner or web camera of laptop. In this proposed system the web camera has been used as QR reader instead of using installing scanner.

b. Security issue: including NIDN inside QR code is more secure than keeping it in plain number.

c. Token requirement: token card can be issued for users including users name and QR to be used as identification card instead of using expensive smart or chip cards. Figure 3-11 (a) illustrates QR coding and (b) illustrates token design.

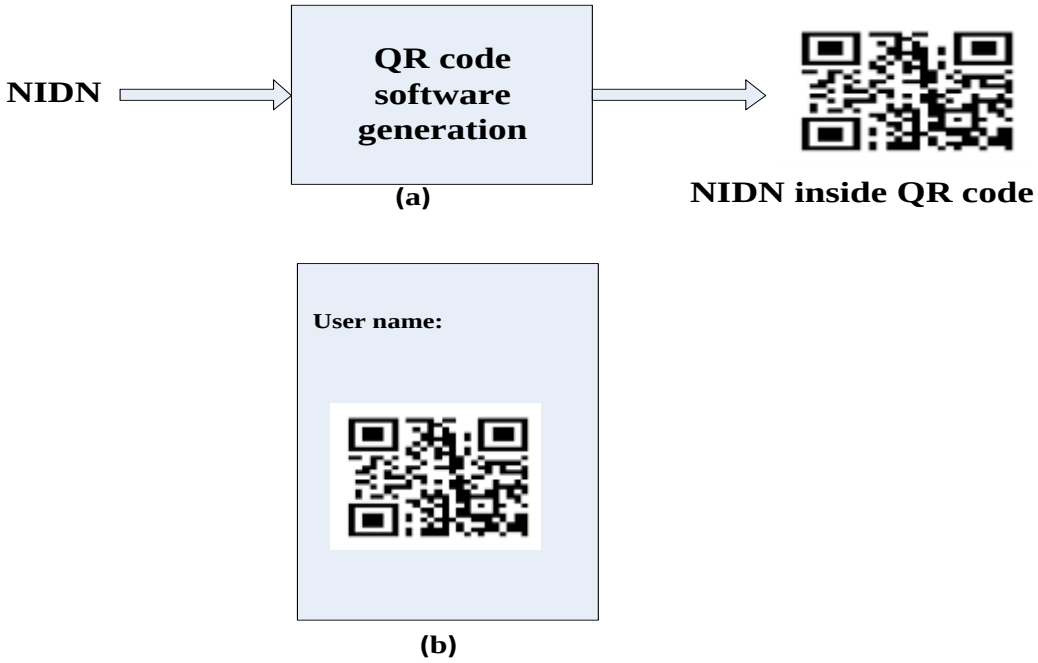

Figure 3-11: (a) QR coding process and (b) user's token form

3.9 QR Decoding

Is the process of retrieving NIDN from QR image. Web camera has been used as QR scanner for QR image capturing. QR decoding software which is a special function has been used for this purpose.

3.10 FP Image Capturing

This module performed in client side, it provides the FP image to the system for registration or authentication requests. FP scanner is configured for compatibility with client programming language and controlled by client program to adjust the image showing and saving for later processing.

3.11 Graphical User Interface

Graphical User Interface (GUI) represents the front-end interface for the system's users. Interaction of users with the system achieved via this module. Simplicity and clarity has been taken into account to guide the users into the system functions. This module provides all the previous function for system's users. GUI of this system is designed and implemented by using C# programming language.

3.12 Client-Server Connection and SSL

The client and server are interacted by using Transmission Control Protocol/Internet protocol (TCP/IP) connection model, which is addressed by IP address and port number. Client-server connection is established during connection negotiation phase while the data exchange during binary streaming with SSL streaming for security issue. Fig. 3-12 illustrates the client-server connection.

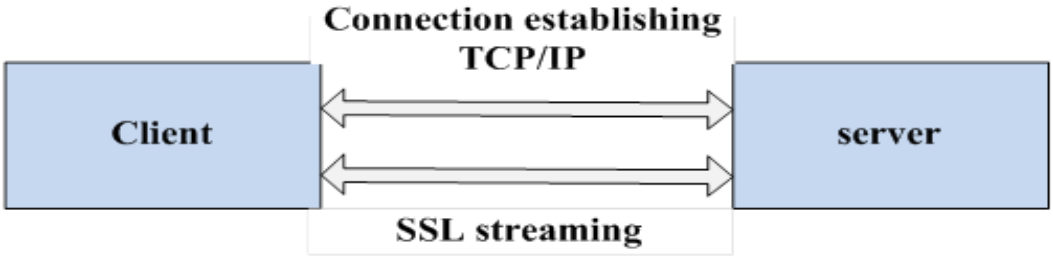

Figure 3-12**:** client-server connection

The protection of connection is an important issue that is used to encrypt the online data. Encryption means that the sender and recipient agree upon a "code" and translate their data. SSL protocol which is described in chapter two is used to create a uniquely encrypted channel for private communications over the public Internet.

Chapter Four

The Proposed System Implementation

4.1 Introduction

Chapter four represents the implementation of the proposed system design (Secure Network Authentication Based on Biometric National Identification Number) described in chapter three. This chapter shows; the implementation tools, system components that implemented by these tools and the results of all system functions. Finally the results discussion also described.

4.2 System Tools

The realization of the system done by using three software tools and one hardware tool, these tools are; MATLAB 8.4, C sharp (C#.NET) 2012, Microsoft Access Database 2010 and ZK 4500 FP scanner, respectively. Figure 4-1 illustrates system's tools interaction.

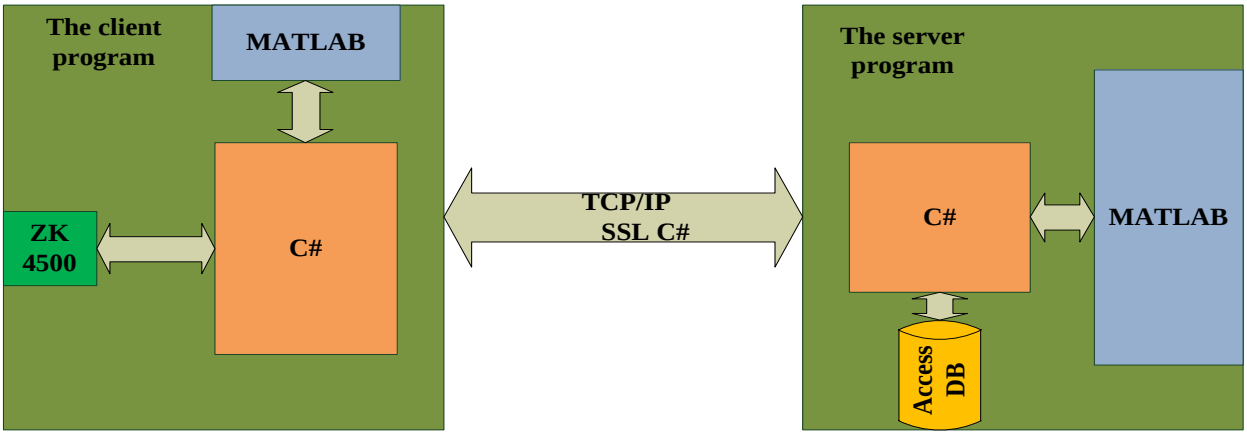

Figure 4-1**:** System's tools

4.2.1 MATLAB

MATLAB is high-level language and powerful interactive tool for numerical computation, visualization and programming. MATLAB can be used for data analyzing, create modules and developing. Wide range of application can be realized using MATLAB like signal processing and communication, video and image processing and etc. [53]. MATLAB is used to implement FP image processing such as enhancement and features extraction, it also used for MD5 implementation, QR coding, QR decoding and QR printing.

4.2.2 C Sharp

C sharp (C#) is a Microsoft's premier language for .NET development application. It leverages time-tested features with cutting-edge innovations, and provides a highly usable and efficient way to write programs for modern enterprise computing environment. It is, considered as one of the most important languages of the twenty-first century [54].

C# is used as interface language to realize the client-server interaction and connection by TCP/IP and SSL, also the end-front of client side implemented by C#. The sever program and it's integration with MATLAB and DB also implemented by using C#.

4.2.3 Microsoft Access

Microsoft access is a Database Management System (DBMS), it provides the tools for DB creation, insertion, deletion and edition. Microsoft access is based on the Sequential Query Language (SQL) [55]. Microsoft access is used to

implement the system's DB which is consisting of one table as described in previous chapter section (3.5).

4.2.4 ZK 4500

ZK 4500 is an optical FP scanner with resolution of 500 DPI. The specification of this device illustrated in the appendix and figure 4-2 illustrates the shape of the ZK 4500 [56].This scanner used to provide the system with the FP image of each user. This scanner integrated within the client program by using interface module developed by C#.

Figure 4-2: ZK 4500 FP scanner

4.3 System Implementation

4.3.1 The client program

The client program represent the GUI to the users of the system, this program dealing with all user's requests by transferring user's data to the server program and gathering the result from the server program and finally presented to the user. Figure 4-3 illustrates the system's GUI that implemented by using C#.

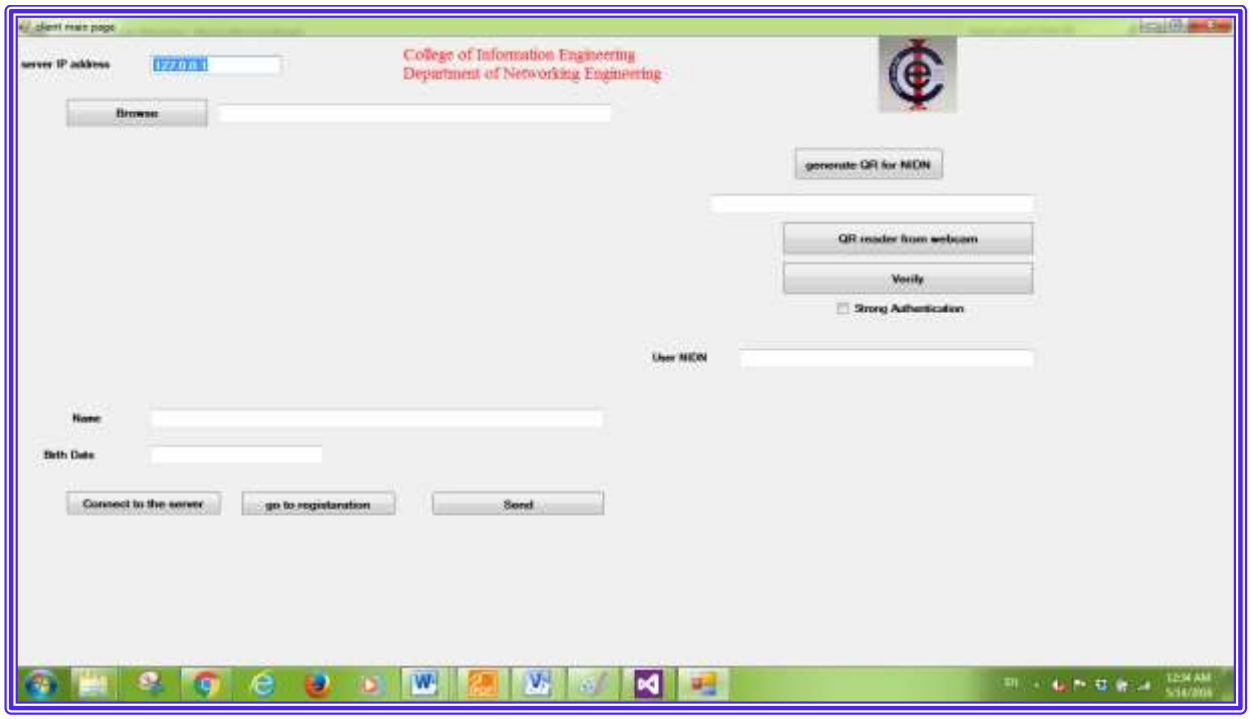

Figure 4-3: Client main program

The client main program contains all functions that provided by the system. The main form contains number of buttons, text boxes to read the user

input and shows output as well as one picture box for FP image. Each button satisfies certain function as follows:

a. **Browse:** this button is used to find the directory of the saved FP image which is used to registration or authentication. This button is used before using the FP scanner or accept saved image in different directories.
b. **Connect to the server:** this button used to establish the client-server connection.
c. **Go to the registration:** this button used to transfer the client to the registration form page for gathering the client information, this is done by clicking this button. Figure 4-4 illustrates the registration form page.
d. **Send:** this button used to send the used information to the server program.

Figure 4-4: Registration form

Registration form contains number of buttons to provide some functions, one picture box to shows the FP image, list boxes to represent the user's birthdate information which are; day, month and year, and two text boxes to read birthdate information and user name. Each button in this form provides the following functions:

i. **Show your birthday in text box:** this button is used to convert the birthdate information from list boxes to the user birthdate text box.
ii. **Start scanner:** this button is used to initialization the FP scanner by sending initial data to the scanner and make it ready for capturing FP images.
iii. **Save FP image:** this button is used to save the FP image that captured by the scanner in the pre-defined directory.

e. **Generate QR for NID:** this button is used to include the NIDN inside QR code.
f. **QR reading from webcam:** this button is used to retrieve the NIDN from the QR code by reading the QR via webcam of the laptop.
g. **Verify:** this button is used to verify request which is either normal or strong. The check box (strong authentication) is used to select the strong authentication request by tuning it true.

4.3.2 The server program

The server program control the whole system's functions such as, registration and authentication by received the data from the client, processes it and return the results. The server program form contain only one picture box to load the received FP image. According to user request, internally the all system functions are achieved such as FP image processing, NIDN generation, QR generation, DB

reading and writing and authentication process. The server program implemented by C# as interface language to be the core of the system, by integrated C# with MATLAB and Access DB to provide the system's functions. Server program tasks distribution illustrated in Figure 4-5.

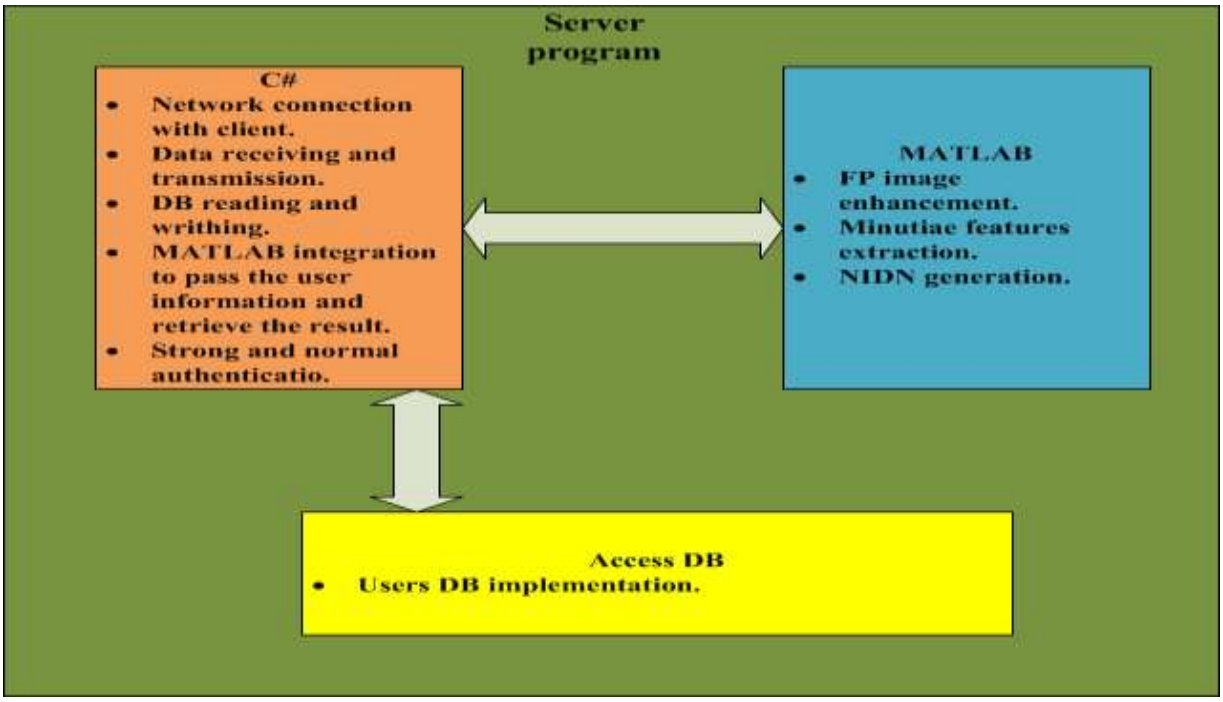

Figure 4-5: The server program tasks distribution.

4.4 User's Registration and NIDN Generation Scenario

This scenario shows the steps of the user registration in the system and NIDN generation for this user. This scenario starts from the user entering his information and end with recording the registration information in DB, these information are; user NIDN, name, birthdate and the FP image path. The user interacts with the client program to perform the registration process. The main steps of this operation are:

a. **Starting:** run the two programs, client and server, as well as connecting the FP scanner via Universal Serial Port (USB).

b. **Connect to the server:** in this phase the client and server has been connected by using C# TCP client and TCP listener with loop back IP address (127.0.0.1) and port number is (1156). This phase done by clicking connect to the server button as shown in Fig. 4-6.

Figure 4-6: Client-server connection operation

After the connection has been done a message box will appears with message "successfully connected" as alert for the connection success.

c. **Go to the registration:** in this phase the user must enter his information for registration, this done by clicking (go to the registration) button to

transfer the user to the registration form. In this situation the user will execute some operations are:

1. Initialize the FP print scanner by clicking the (start scanner) button as shown in Fig. 4-7.

Figure 4-7: FP Scanner initialization

Message box will appear with a message of "initial success" as alert for successful initialization of the FP scanner to be ready for capturing.

2. Enter the user information like name, birthdate and FP of the fore finger of the right hand by putting the user's finger on the surface of the scanner as shown in the image in the figure above. Figure 4-8 illustrates the user's information registration and FP image capturing. At this stage the user's information are loaded in the

registration form. By clicking the exit button in the lower left side of the registration form, all user information will be automatically transmitted to the main form as shown in figure 4-9.

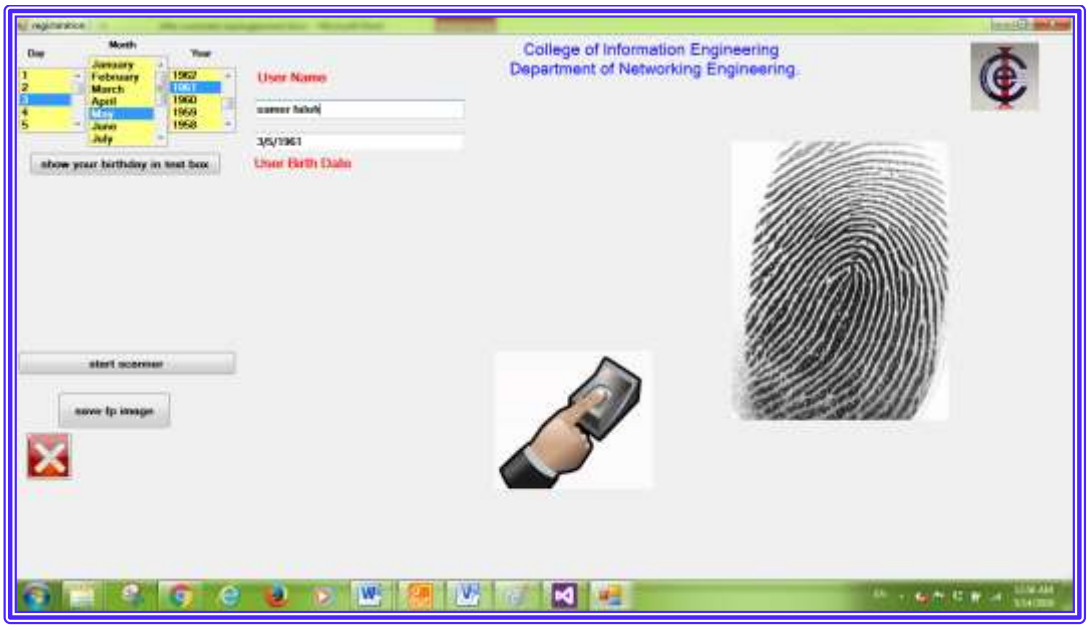

Figure 4-8: User registration

Figure 4-9: user information in the client main form.

d. Sending: in this phase the user's information will be sent to server program by clicking the (send) button. The sending of this information done in secure form to be protected from any sniffing attack by using SSL.

At this point the user information has been transmitted to the server program as shown in figure 4-10. The server program will start processing the information to generate the user's NIDN and recording in the DB.

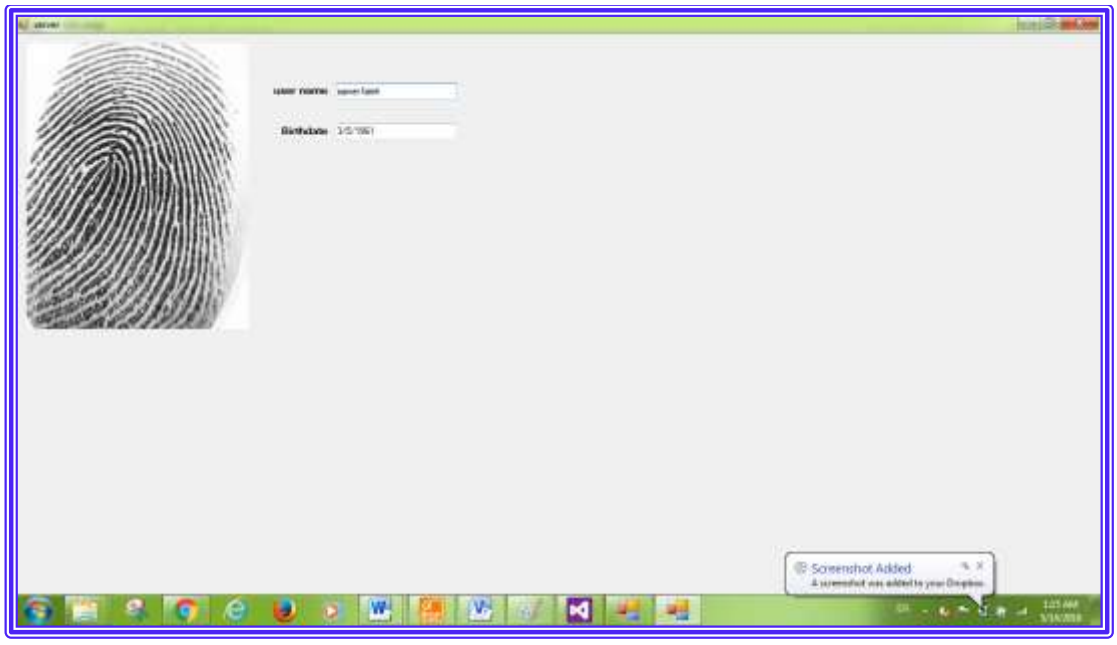

Figure 4-10: user's FP at the server program

4.4.1 NIDN Generation

The generation of the NIDN has been done in the server by using MATLAB. C# program passes the user information (FP image and personal information) to the MATLAB program. The processing steps in MATLAB program for NIDN generation are:

a. **FP image preprocessing:** in this step the FP image passed through the stages of the FP image enhancement algorithm to get best quality image, after completion of the enhancement the binarization and thinning are applied for the enhanced image. The sub-steps of FP image enhancement from input image to output image described as follow:

   i. **FP image segmentation and normalization:** in this process the input FP image segmented and normalized with zero mean (M = 0) and unity standard deviation ($\sigma = 1$) as shown in figure 4-11.

Figure 4-11: FP image normalization.

ii. **Finding the orientation image:** in this process the orientation estimation has been done and the orientation image has been generated as shown in figure 4-12.

The frequency of the local ridge is calculate by using the method that described in chapter two, the value of the frequency is used in adjustment of Gabor filter transfer function.

Figure 4-12: FP orientation image.

iii. **Gabor filtering:** in this step the normalized FP image will convolve with GF transfer function that is tuned with the required specification (ridge orientation, ride frequency, mean value and stander deviation). Figure 4-13 illustrates GF applying output.

Figure 4-13: GF output.

iv. **Binarization:** the enhanced FP image will convert from gray level (0-255) to binary level (0-1) by using MATLAB binarization function with threshold of (-1) as shown in figure 4-14.

Figure 4-14: Binary FP image.

v. **Thinning:** the enhanced binary FP image is thinned to be with one pixel ridge width. This operation has been done by using a binary morphological function called "thin" which is supported by MATLAB code as "`thinned= bwmorph(img,'thin',Inf);`". Figure 4-15 illustrates the thinned image.

Figure 4-15: Thinned FP image

b. **FP minutiae marking and extraction:** in this process the CN algorithm has been applied to the thinned FP image for minutiae marking (edge termination and bifurcation) as shown in figure 4-16. The minutiae detection located the minutiae position, type and orientation. The minutiae matrix also created to contain of the detected minutiae features of the input FP image. The minutiae matrix can be saved as a feature vector.

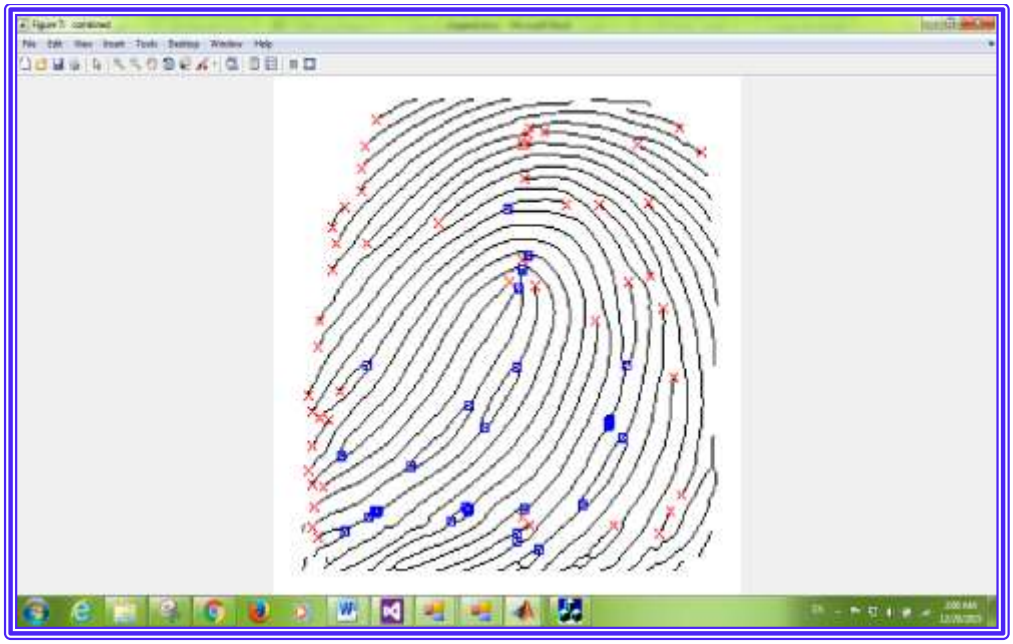

Figure 4-16: FP minutiae detection

For located minutiae in figure 4-16, the count of minutiae is 81 between edge termination and bifurcation. The matrix size is 4x81which is very long to present, so take the first twenty minutiae as shown in table 4-1. The minutiae type represented with values of "1" specify the ridge termination and "3" specify the ridge bifurcation.

3. **Combining the minutiae features with personal information:** At this point the minutiae matrix is converted to a string as show in figure 4-17.

Table 4-1: Minutiae matrix

X	Y	Type	Orientation
022	057	1	2.5254
026	141	1	2.9421
026	223	1	0.5128
028	149	3	3.0977
029	140	1	2.9355
034	137	1	2.8721
034	139	1	2.9118
034	200	1	0.4588
035	051	1	2.3969
038	235	1	0.6912
046	049	1	2.3598
051	138	1	2.8900
057	049	1	2.3844
063	206	1	0.7387
064	161	1	0.4074
064	179	1	0.6807
065	040	1	2.2792
066	129	3	2.8126
073	091	1	2.4617
075	033	1	2.0349
083	035	1	2.1236

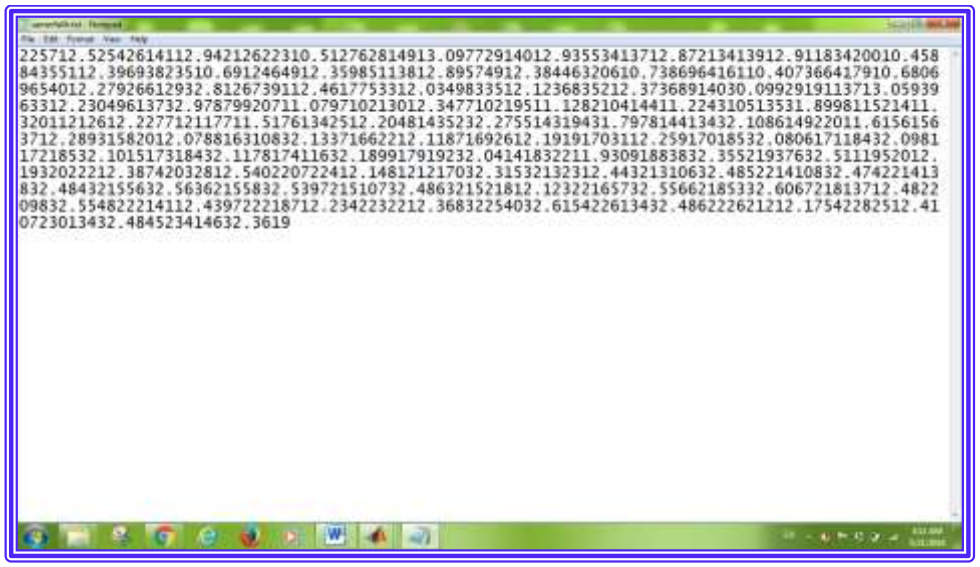

Figure 4-17: Minutiae string

After the string generation, minutiae features string is concatenated with user information string (name and birthdate). In this case of registration the user name is “Samer Faleh” and his birth date is “3/5/1961”. The produced combined string illustrated in figure 4-18.

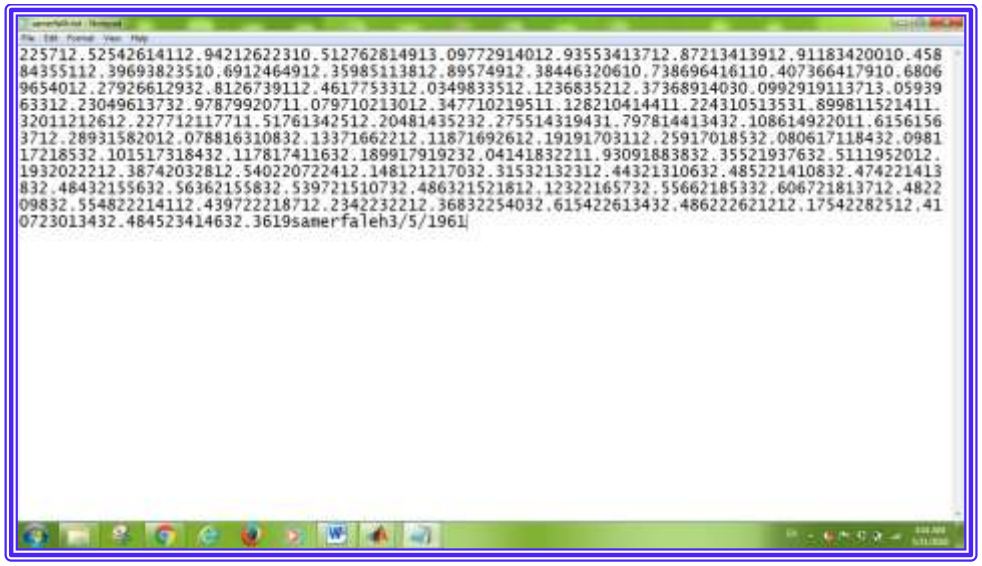

Figure 4-18: Combined strings

d. **Applying MD5:** in this process the combined string will be entered to the MD5 hashing function to produce fixed length 128-bit hash value in the form of 32 hexadecimal digits. For this case the NIDN is "35bdf7e7ffdf077def3b8967d28ed2a9" and returned by message box as shown in the figure 4-19.

4.4.2 Recording in the DB

After the completion of the NIDN generation the server program will record the user information in the DB. The user record in DB contains; the user NIDN, name, birthdate and FP image saved in Bitmap (BMP) image format for future using in the authentication process. Figure 4-20 illustrates the user's records in the DB.

Figure 4-19: NIDN

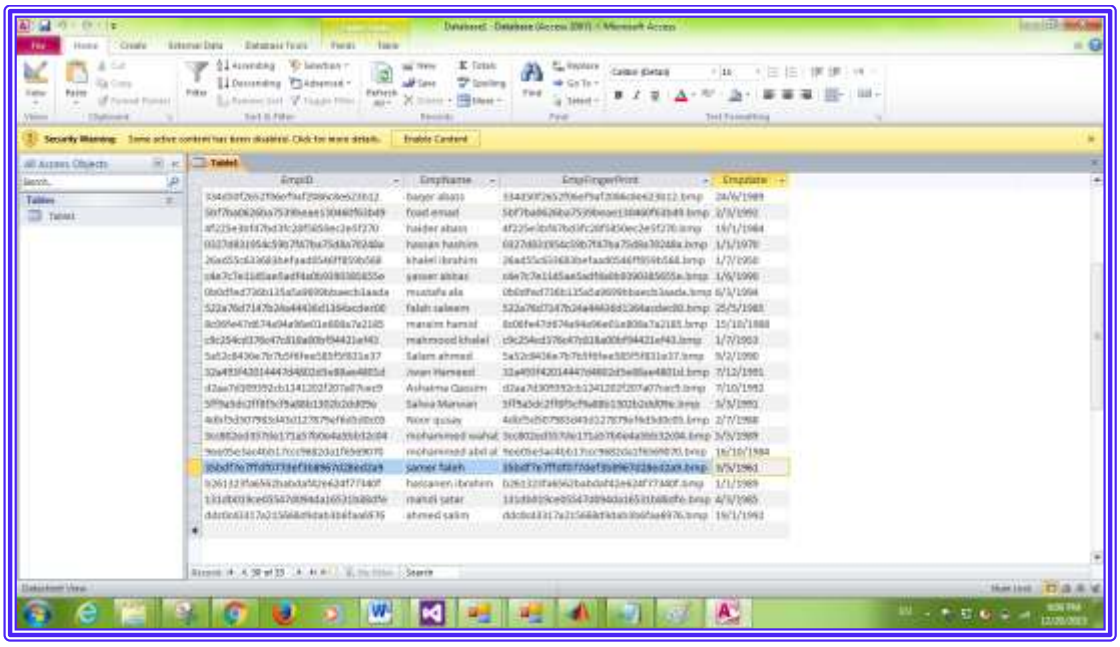

Figure 4-20: user recording in DB

4.5 QR Generation

After completion of NIDN generation the server will include the NIDN in QR image as shown in figure 4-21. The QR is generated to be used as token readable card. For QR generation, the user must click the (generate QR for NID) button in the main form of the client program, the server program will call the software of the QR generation. A special software library called (ZXing library) which is image processing software for multi format one-dimension barcode and two-dimension QR implemented in Java with ports to other languages. In this research this library has been used with MATLAB, the input value is NIDN, the output is QR image including NIDN in form of MATLAB figure as shown in figure 4-22.

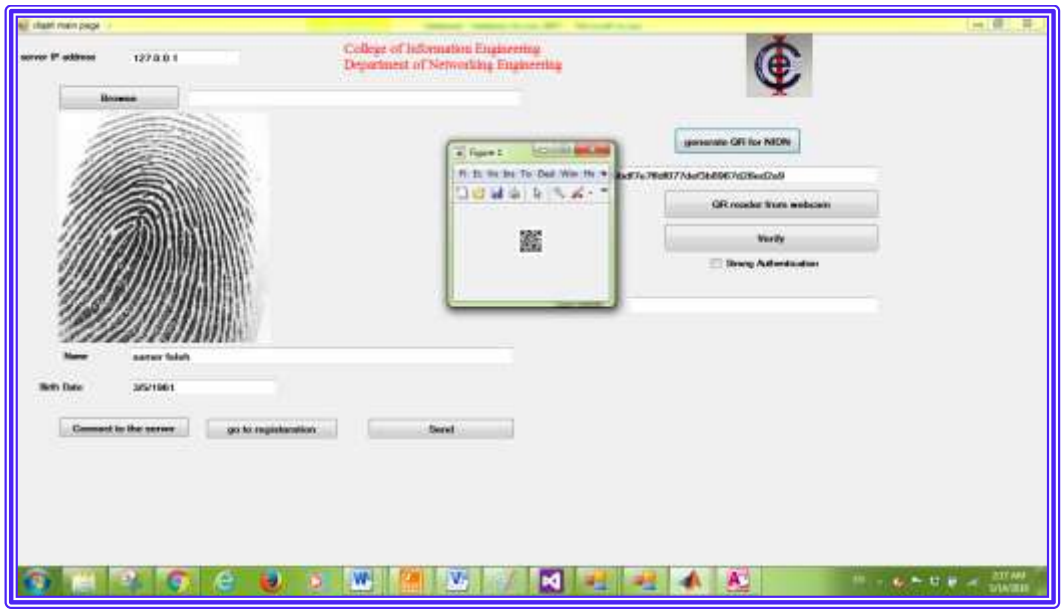

Figure 4-21: QR generation

Figure 2-23 shows the produced QR full image.

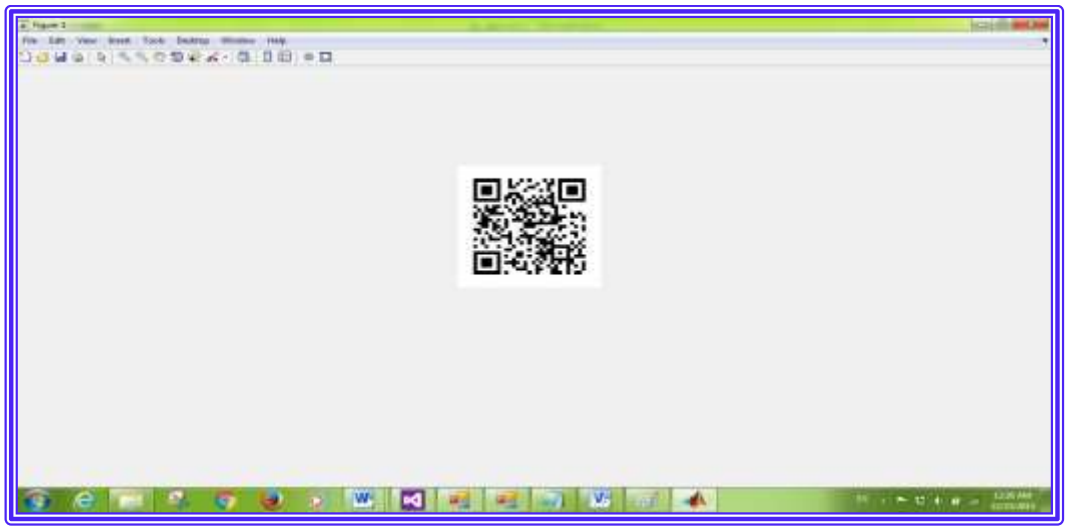

Figure 2-22: QR image.

The token card is generated by printing the QR image in the middle of appropriate size white paper with the user name and plaintext NIDN if needed.

4.6 Normal Authentication Scenario

This scenario shows the normal authentication process. This process starts with presenting user's token card and end with return the result of user's existence in system DB. Normal authentication includes; QR decoding, NIDN sending to the server, checking the received NIDN with the all NIDNs in the DB and finally return the result of checking which is either the NIDN is found or not. For this system fourteen users have been registered in system DB.

In QR decoding step the NIDN will be retrieved from the QR code by clicking button (QR reader from webcam) in the client main form. QR image in the token card will be captured by web camera of the lab top, the captured QR image will be sent to QR decoding software. The same software library that used in QR generation (ZXing library) is used but with decoding mode. Decoding mode take a QR image and return the content that has been included in this image. The result of QR decoding is NIDN that included inside QR image appears in (user NID) text box in the client main form. Figure 4-23 illustrates the QR image capturing process.

Figure 4-23: QR image capturing

Figure 4.24 illustrates the QR image decoding to retrieve the NIDN which will be used in the normal authentication process.

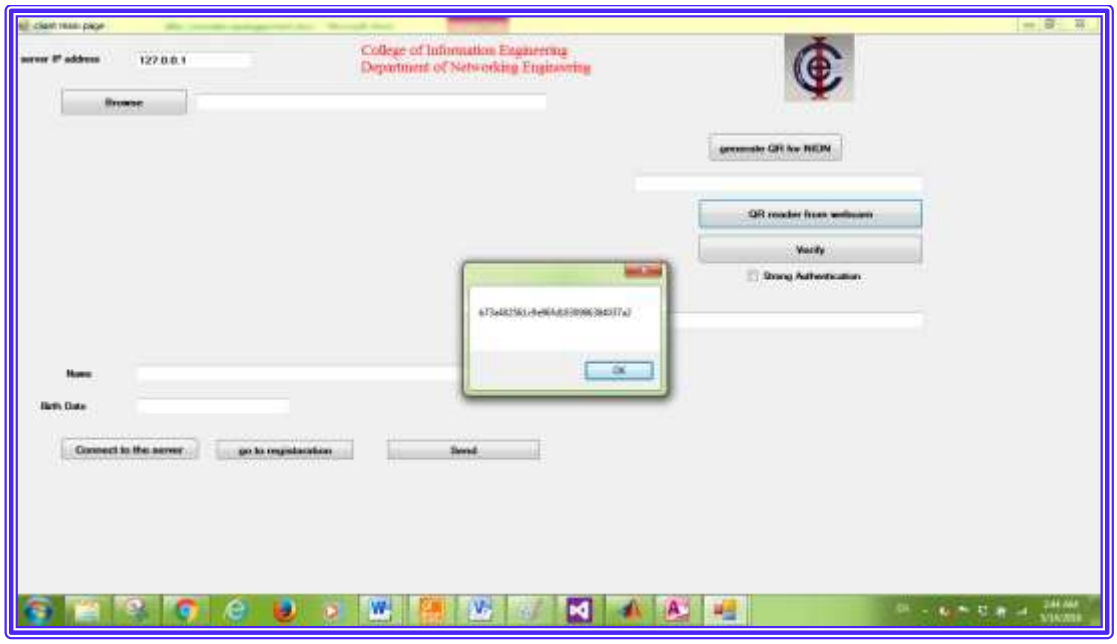

Figure 4-24: QR decoding

The message box appears in figure 4-24 contains the retrieved NIDN with value of "b73a482561c9e96fd1830986384037a2", which is NIDN of the user "mntaser saleem" generated in registration phase. The normal authentication process begin by clicking the button (verify) in the main form of the client program without selection the check box (strong authentication) this selected in the strong authentication only. The NIDN will be sent to the server program in secure format (SSL), server program will compare this NIDN with others NIDN in system DB. If the comparing result is true, the user has been recorded and can access to public applications, else the user is not found and denied from public application. In successful normal authentication as in this example, message box will appear with a phrase of (the user is found in DB) as shown in figure 4-25 as alert to existence of user NIDN in DB, after this the user automatically transfer to the application web page as shown in figure 4-26.

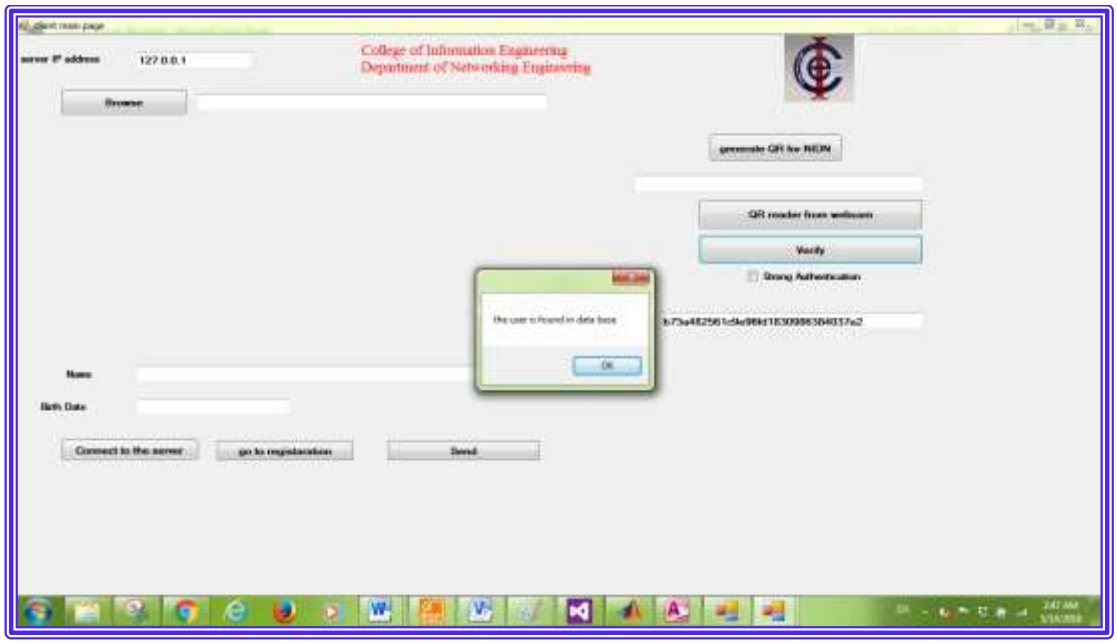

Figure 4-25: Success normal authentication

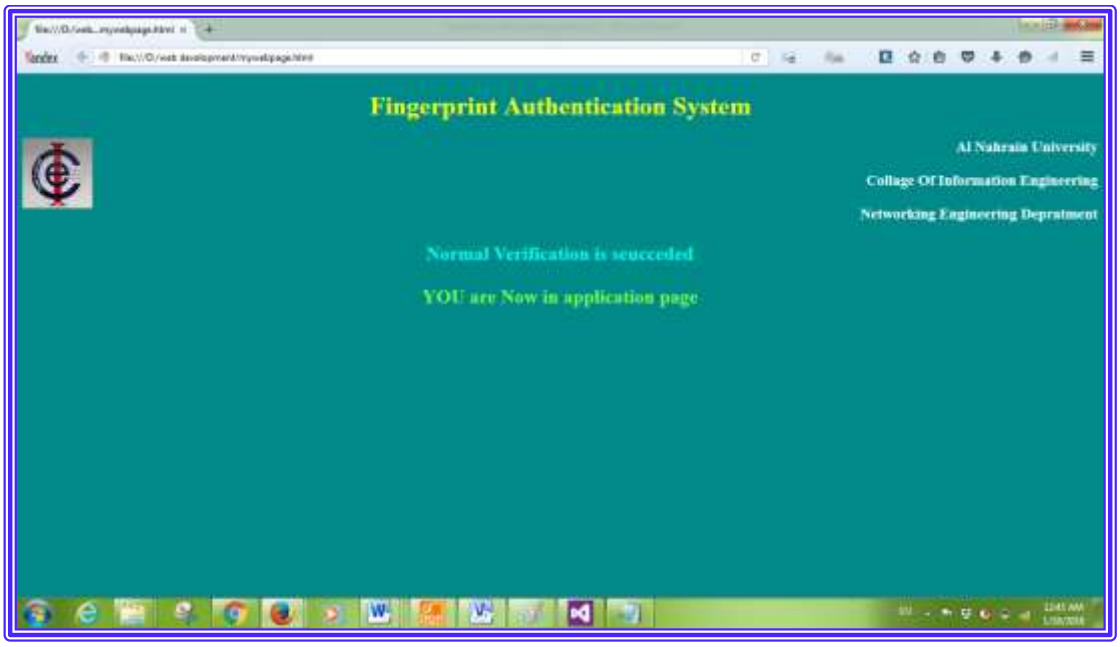

Figure 4-26: Normal application web page

4.7 Strong Authentication Scenario

The second authentication service supported by the system is strong authentication. Strong authentication is used to protect sensitive information. In strong authentication the user must introduce his token card and his FP. The NIDN will use as identifier for the user and the FP will use as verifier. The matching score value will identify the user acceptance or rejection for sensitive application authentication. Strong authentication early operation steps are similar with the normal authentication such as QR reading, QR decoding and comparing the NIDN with the saved NIDNs in the DB. The same user in normal authentication with NIDN is “b73a482561c9e96fd1830986384037a2” has been accessed strong authentication service. Strong authentication phases are:

1. **QR capturing and decoding phase:** as in normal authentication, the user will introduce his token card to retrieve the NIDN from QR image.
2. **Strong authentication request phase:** the user must click the (verify) button in the client main form and select check box (strong authentication). The result of this action is message box with text “for strong authentication please present your fingerprint” to alert the user for capturing his FP as shown in figure 4-27.

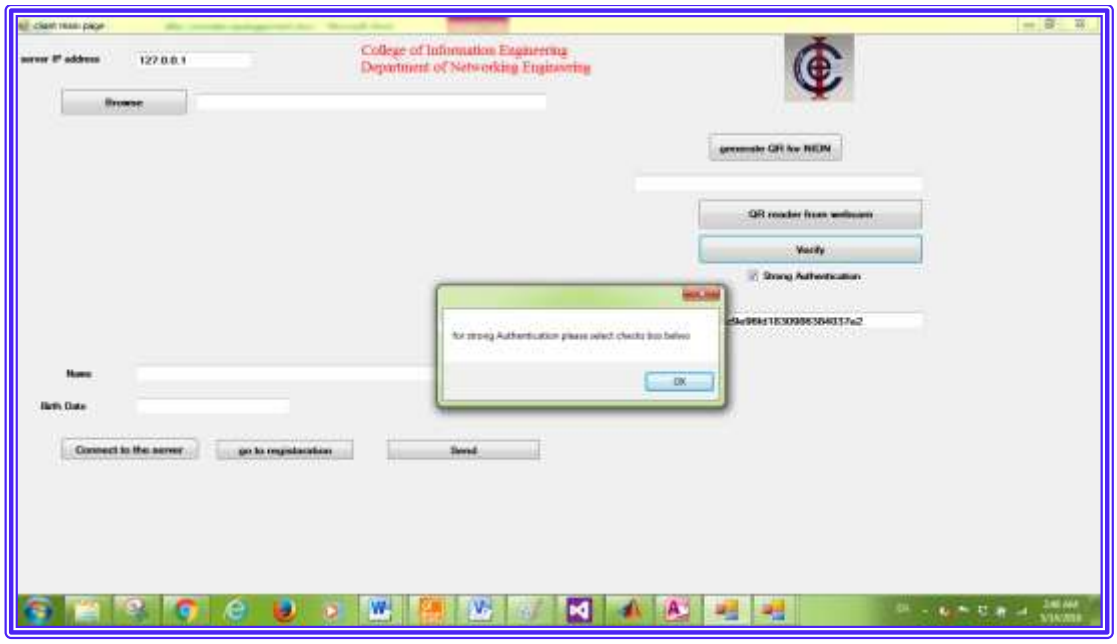

Figure 4-27: Strong authentication request

After this point the user will be transferred to the registration from for FP image capturing. Figure 4-28 shows FP image capturing for strong authentication request.

Figure 4-28: FP image capturing in strong authentication

After this situation the FP image automatically transfer to the main form by clicking (exit) button. The FP image and the retrieve NIDN will be sent to the server program in secure format using SSL protocol, in the server the system will compare the NIDN with others NIDN in system DB. In this scenario the same user of the normal authentication is accesses to strong authentication, so that the result of NIDN comparison is true with messages box "the user is found in data base" as shown in figure 4-29.

Figure 3-29: User identification in strong authentication

After user identification, the system will retrieve the saved FP image from system's DB. The two FP images will enter to the matching sub-system for user verification.

Matching sub-system is a software of FP recognition engine described in the previous chapter, this sub-system take two FP image in Bitmap (BMP) format and return matching score value which is float data type value. The decision of the user verification is depends on the matching score value, if score is greater than threshold value) the user is accepted, else the user will be rejected from the sensitive applications (private services). In this example the matching score is (77.73013) between the two FP image as shown the message box in figure 4-30 and threshold is (50). The threshold value of (50) has been selected after trade off among number of matching algorithms and displays the middle state between user's rejection and acceptance rate.

Figure 4-30: Matching score value

The condition of the user's acceptance is satisfied, system will alert the user in message box of "strong authentication is true and you can access to the private- applications" and transferred the user to the private application page as shown in figure 4-31.

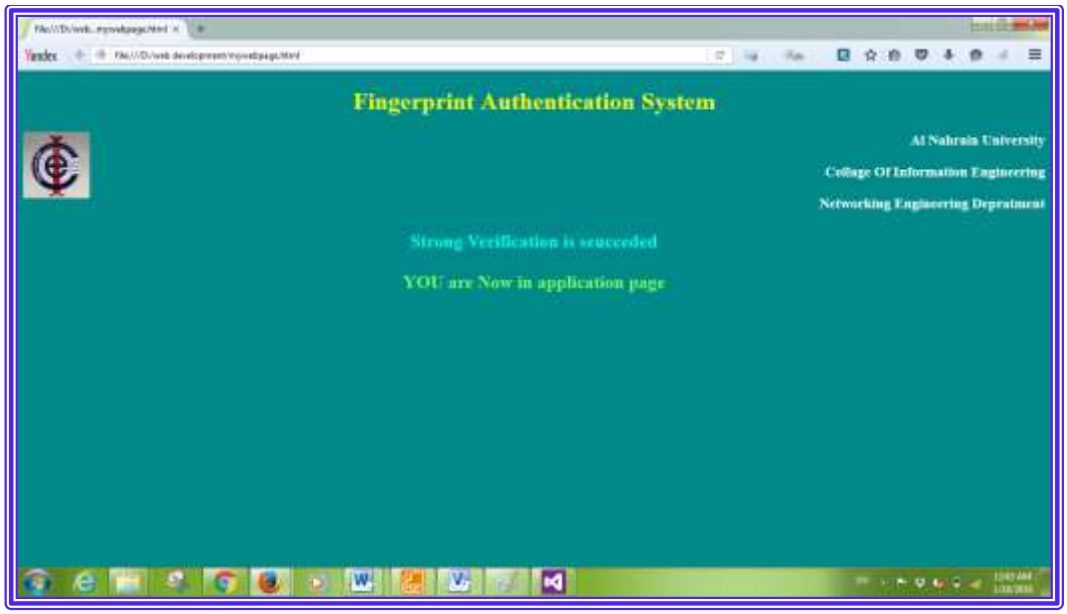

Figure 4-31: Strong authentication web page

2.8 Faked Access Scenario

This scenario shows the case of steeling the token card of a recorded user and using it by an unrecorded user to access the private application. As in the previous scenario the token card with information are; user name is "mntaser saleem" and the NIDN is "b73a482561c9e96fd1830986384037a2", has been used in this test. The first operations such as QR reading, QR decoding and NIDN comparing are similar to the strong authentication scenario first steps with same output so there is no needed to repeats it. The different result will appear in FP matching between the saved FP image and the online impressed FP.

The matching score value of this case is (0) as shown in figure 4-32, because the two FPs are mainly different (from different persons). According to this score value the system will reject this request from system access (because the matching- score less than threshold value) by responding in message box with phrase of "verification failed" as shown in figure 4-33.

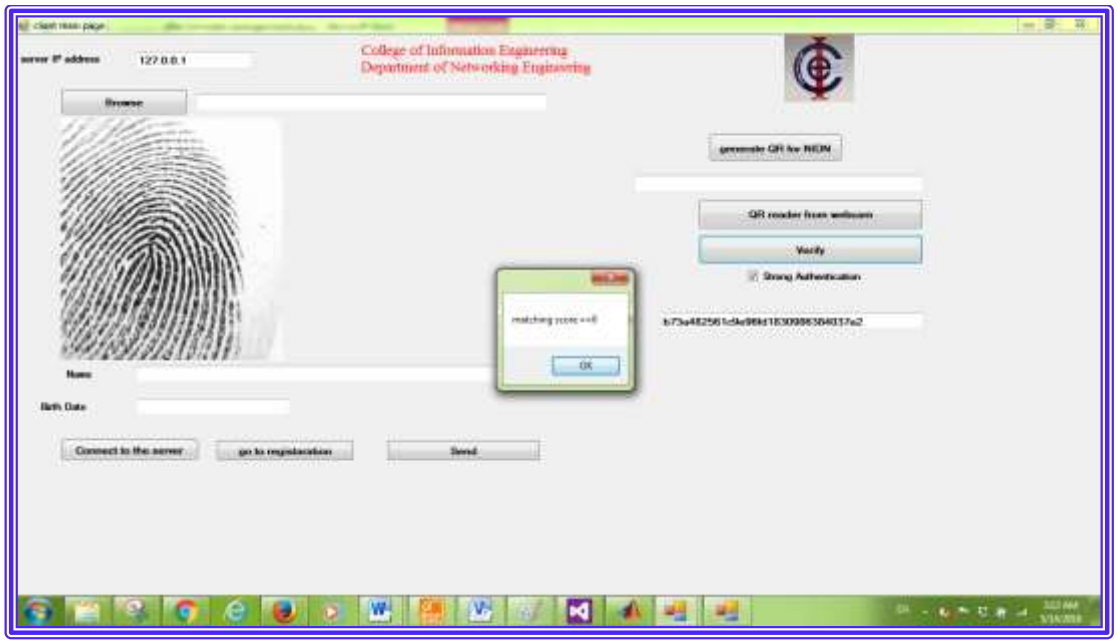

Figure 4-32: Matching score value in faked strong authentication

Figure 4-33: Strong authentication rejection.

4.9 system testing

4.9.1 Matching sub-system testing

Mainly, any authentication system that used any types of biometric depends on the recognitions system, in our research the matching sub-system is FP matching (AFIS engine). This system should be tested to evaluate its performance in FP matching. There are number of testing method for performance evaluation such as matching score value at different PF impression conditions and the recognition percentage rate for access control.

a. Matching Score at Different Conditions

This assessment shows the FP matching score value at different conditions of FP impression on the surface of scanner, these different conditions are;

forward shifting, backward shifting, clock-wise rotation and anti-clock-wise rotation. Figure 4-34 reveals the FP impression conditions.

Figure 4-34: (a) FP backward shifting, (b) FP forward shifting, (c) FP clock-wise rotation and (d) FP anti-clock-wise shifting.

After applying these FP impression we get the matching score result as in table 4-2.

Table 4-2: FP matching sub-system testing result.

FP Impression condition	Matching Score value
Backward shifting	55.95922
Forward shifting	65.58824
Clock-wise rotation 45°	42.90625
anti-clock-wise rotation 45°	67.27821

The FP matching between the saved FP image and real time captured FP is mainly dependent on some condition such as; using dirty fingers and experience of using the FP scanner device. The experiment of accessing an expert user to the system is matching score value of 94.15709 as shown in figure 4-35, this result can increase or decrease according to above conditions.

Figure 4-35: Expert user matching score value

b. Recognition Rate

This test presents the percentage recognition accuracy of the FP matching-sub system. This test depends on acceptance the wanted users and rejection the unwanted users relatively to all accessed users. Calculation of this test achieved as follow:

1. Access number of recorded users to strong authentication, count the number of successful verification and failed verification.
2. Access number of unrecorded user within NIDN of recorded user to strong authentication, count the number of successful verification and failed verification.
3. Take the percentage average value as follows:

   Recognition rate (%) = $\frac{SA}{TU}$100%.

   Where **SA:** successful matched count, **TU:** total number of submitted users.

Recognition rate test is mainly depends on the threshold value, the ideal threshold value of (AFIS) is (0), this is also tested by trying number of unrecorded user as a result the score is zero for all experiments. Zero score means no matching between the two FP. At threshold of (0) the FAR and FRR are crossed (intersected). For 26 users, 13 recorded users and 13 unrecorded users. The true and false matching values are calculated. Table 4-3 shows the result of this calculation. The recognition rate for the result in table 4-4 with different threshold value is presented in the following table.

Table 4-3: Results of matching

User Name	User NIDN	True Matching Score	False Matching Score (faked)
Mntaser Saleem	b73a482561c9e96fd1830986384037a2	93.22698	0
Mustafa Taher	35bdf7e7ffdf077def3b8967d28ed2a9	77.22011	0
Rafel Saleem	da612dafcc144ca10334add17b6dba66	72.1989	0
Mustafa Ali	922fca9f9f70f8f850f7f0cfb12dfc54	40.95295	0
Samer Faleh	35bdf7e7ffdf077def3b8967d28ed2a9	75.94591	0
Mahdi Satar	131db019ce05547d094da16531b88dfe	58.06531	0
Mokhaled Saleem	ff7bcafb6eebc6ad89243bbdf5cf18df	99.31831	0
Mustafa Haji	5f0c73aa1caccd272cca3c4442842e5d	70.96764	0
Dhia Raad	e5f0555cf991d544696808d8eacbad17	99.56341	0
Foad Emad	5bf7ba0626ba7539beae130460f63b49	54.16734	0
Hasanen Ibrahim	b261323fa6562babdaf42e624f77340f	55.6778	0
Sabah Faleh	3bd27e91ece2697d1bb10f43bf4711f5	50.86031	0
Zia Raad	5934a3420ee240fff4c53001cd710ea1	76.24079	0

Table 4-4: Recognition rate percentage

Threshold value	Recognition rate (%)
0	100%
10	100%
20	100%
30	100%
40	100%
50	96.153%

4.9.2 Traffic Analysis

This test shows the traffic of data transferring between client- server interactions to assess the function of SSL protocol in security. This test has been achieved by using of software tool for packet sniffing called (Wire-Shark). Two cases are presented:

1. **Data transmission without using SSL:** This case presents client-server data transmission without using SSL. This lead to problem of capturing user's data by any eavesdropper and can retrieved the plain text because there is no encryption method has been used to protect the application data. Figure 4-36 shows packet capturing without SSL (without encryption)

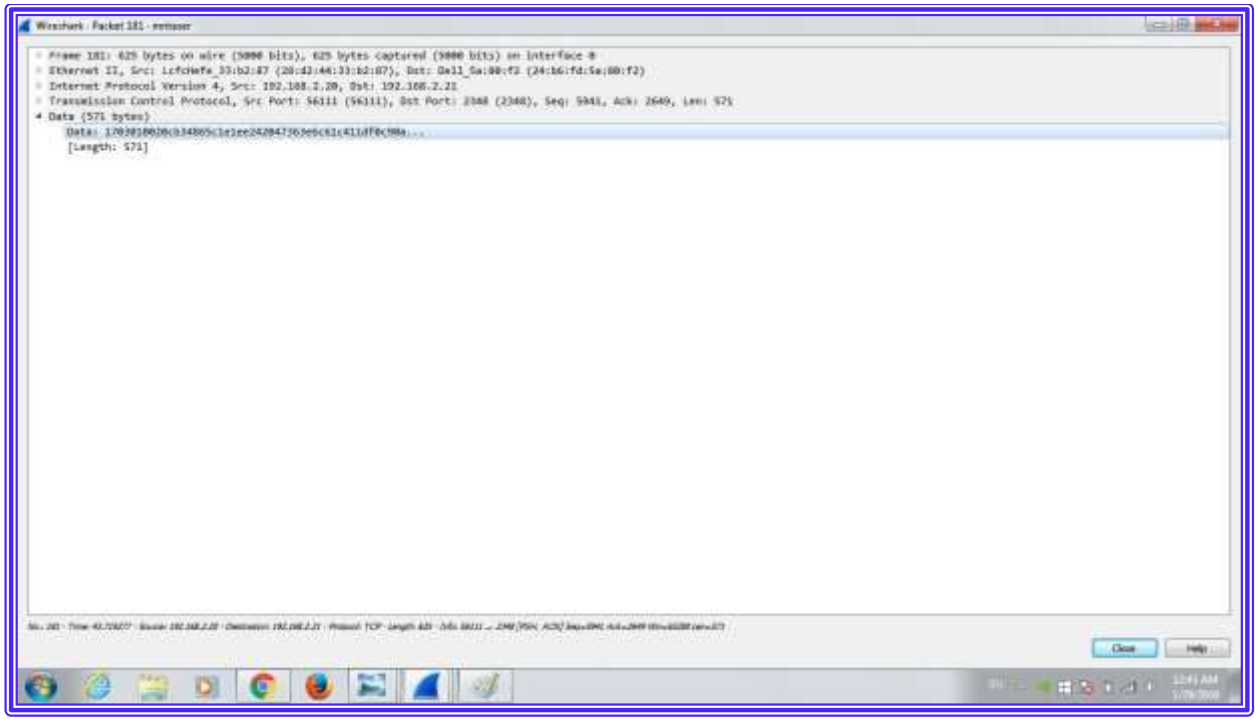

Figure 4-36: user application data without SSL

2. **Data transmission using SSL:** This case presents client-server data transmission by using SSL, this prevent any eavesdropper attack from retrieving the plain text unless analyzing the encryption algorithm and get the secret key but this needs additional effort and time. Figure 4-37 shows packet capturing with SSL.

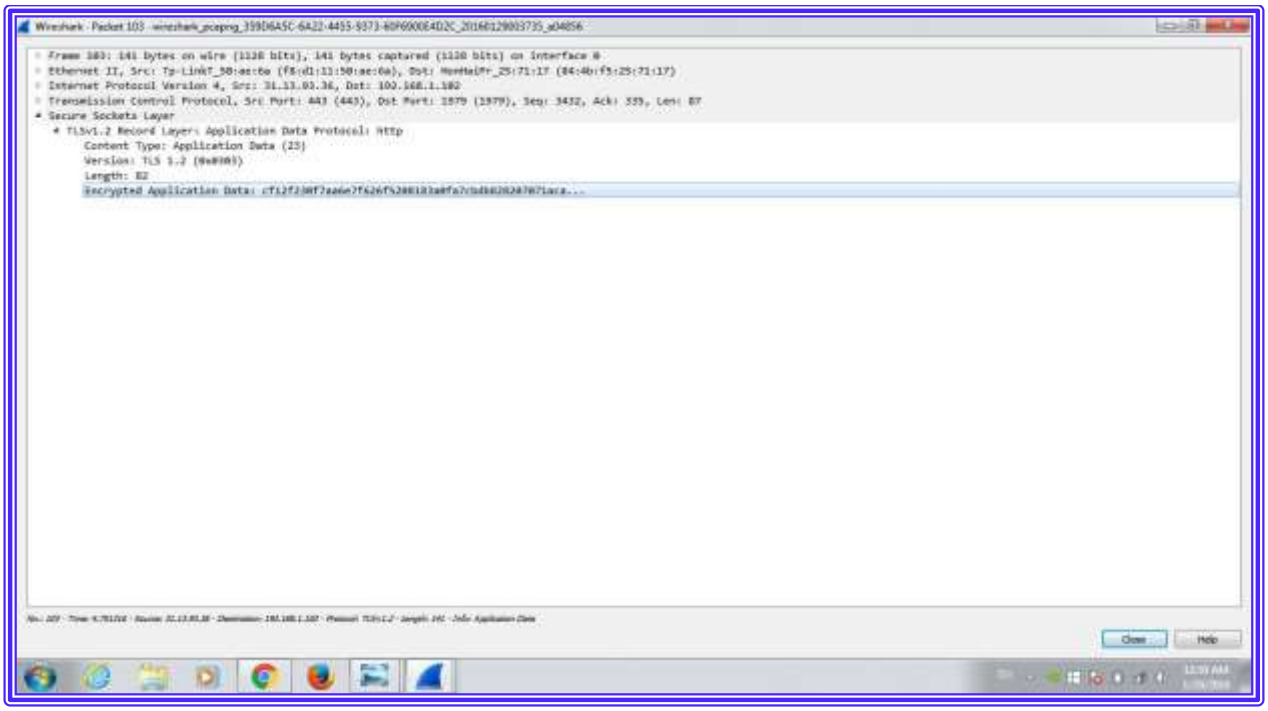

Figure 4-37: packet capturing with SSL.

Finally, executed time of some system's processes has been calculated Table 4-5 presents these results.

Table 4-5: some processes execution time.

process	Time in second
Gabor filtering	6.492648
Binarization	0.217387
Thinning	0.159063
Minutiae extraction	1.289938
String generation	14.129950
MD5	0.903517
NIDN generation	23.838873
QR generation	0.578044
QR decoding	0.152133

4.10 Result Discussion

Producing matching score of (0) between the different FPs is related to the FP matching sub-system (AFIS) that has been used. The purpose of tunes this value to (0) is eliminate FAR increasing and makes the (0) as ideal threshold value (FAR and FRR intersection point).

With threshold values great (0), recognition rate may be less than (100%) because of rejection in an authorized users. Increasing in FRR is related to the decreasing matching score values of authorized users because some condition such as using dirty finger and less experience in using FP scanners, this makes a big different with the saved FP (template).

Gabor filtering is good FP enhancement algorithm with accepted executing time of (6.492648 sec), quality of enhanced image has been appeared in minutiae extraction. Minutiae extraction algorithm (CN) is mainly depends on the quality of FP image, CN executing time is also accept (1.289938 sec).

The hashing output of MD5 is very suitable (128-bit) 32 hexadecimal digits, on long and coverage a wide range of output (2^{128}). The execution time of MD5 is accepted (0.903517 sec).

Chapter Five

Conclusions and Future Work

5.1 conclusions

Through the system design and implementation phase, some conclusions are drawn, these are:

1. Strong authentication service is a good method to protect the private application that contains sensitive data from access by unauthorized users, but will cost more computation time.
2. Some application such as electronic libraries (e- libraries) and billing payment systems needs a normal authentication method with light weight computation efforts, so the normal authentication is a more appropriate for this purpose.
3. Including NIDN in QR image is more feasible to be secured and fast retrieved by capturing device during system access comparing with using the plain NIDN.
4. SSL is a good security protocol in provided a secure data transferring between the client and server sides.
5. Using of time domain Gabor filtering method for fingerprint quality enhancement is more appropriate enhancement method for minutiae features extraction algorithms.
6. A minutia features are a good features in comparing among fingerprints and differ from one person to another but suffering from instability against scaling and rotation of fingerprint.

5.2 Suggestion for Future Work

Several suggestions are identified that could be implemented in the future to develop the proposed system:

1. Combine more than one biometric type in NIDN generation to make the NIDN more secure for example combing FP with iris.
2. Develop a complex algorithm to solve the problem of the stability in FP features extraction and generate same NIDN in each time, for example make a refinement for the primary minutiae features to generate a stable secondary features that resist the fast change in primary features.
3. Develop a memory card with high capacity to store all fingerprint features for a user to be used in other authentication process without necessity for central DB.
4. Using of password mechanism to create a password for each NIDN during registration phase. In this case each user has a NIDN and password. Using of both is feasible to protect the user's right in case of stealing user's NIDN token and presents it for system access by an unauthorized person.

References

[1] A. Jain, L. Hong and S. Pankanti, "**Biometric Identification**"

Communications of The ACM, Vol. 43, No. 2, February, 2000.

[2] A. K. Ojha, "**ATM Security using Fingerprint Recognition**", International Journal of Advanced Research in Computer Science and Software Engineering Research Paper, Vol.5, No.6, June 2015.

[3] A. Jain, A. Ross and S. Prabhakar, "**Fingerprint Matching Using Minutiae and Texture Features**", Int'l Conference on Image Processing (ICIP), pp. 282-285, October 2001.

[4] C. C. Ho and C. Eswaran, "**Consodilation of Fingerprint Databases: A Malaysian Case Study**", 11th International Conference on Hybrid Intelligent Systems (HIS), 2011.

[5] L. Hong, Y. Wan and A. Jain, "**Fingerprint Image Enhancement: Algorithm and Performance Evaluation**", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 8, August 1998.

[6] R.Thai, "**Fingerprint Image Enhancement and Minutiae Extraction**", PhD. Thesis, University of Western Australia, 2003.

[7] I. Babatunde, A. Kayode, A. Charles and O. Olatubosun, "**Fingerprint Image Enhancement: Segmentation to Thinning**", International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 3, No. 1, 2012.

[8] N. Negi and S. Semwal, "**Minutiae Extraction and Pruning Based Fingerprint Identification with Pattern Classification by Radial Basis Function**", International Journal of Engineering Research and Applications (IJERA), Vol. 3, No. 4, August 2013.

[9] B. Ne'ma and H. Ali, "**Multi-Purpose Code Generation Using Fingerprint Image**", The International Arab Journal of Information Technology, Vol. 6, No, 4, October 2009.

[10] I. Jabber, "**Authentication Method Based on Hashes Authentication Fingerprint For fast retrieval**", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Vol. 1, No. 1, May-June 2012.

[11] H. Salman, "**Proposal Design: Fingerprint Random Number Generators**", the 13th International Arab Conference on Information Technology, December, 2012.

[12] S. Ambadiyil, K. Soorej and V. Pillai, "**Biometric based Unique ID Generation and One to One Verification for Security Documents**", International Conference on Information and Communication Technologies (ICICT), 2014.

[13] P. Zhang, X. Guo and J. Gadedadikar, "**Online Fingerprint Verification Algorithm and Distributed System**", Journal of Signal and Information Processing, pp. 79-87, 2011.

[14] Y. Li-qiang and G. Ling, "**Feature Extraction of Fingerprint Image Based on Minutiae Feature Points**", International Conference on Computer Science and Service System, IEEE, 2011.

[15] N. Bhargava, R. Bhargava, M. Mathuria and M. Cotia, "**Fingerprint Matching using Ridge-End and Bifurcation Points**", International Conference in Recent Trends in Information Technology and Computer Science (ICRTITCS), 2012.

[16] M. Lourde R and D. Khosla, "**Fingerprint Identification in Biometric Security Systems**", International Journal of Computer and Electrical, Vol. 2, No. 5, 1793-8163, October,2010.

[17] A.K. Jain, L. Hong, S. Pankanti and B. Bolle, "**An Identity-Authentication System Using Fingerprints**", Processing of The IEEE, Vol. 85, No. 9, September 1997.

[18] O. Aiy, H. Onsi, G. Salama and T.Mahmoud, "**Multimodal Biometric System using Iris, Palmprint and Finger-Knuckle**", International Journal of Computer Applications, Vol. 57, No.16, 0975 – 8887, November, 2012.

[19] M. Al-Hassani, A. Kadhim and V. Samawi, "**Fingerprint Identification Technique Based On Wavelet-Bands Selection Features (WBSF)"**, International Journal of Computer Engineering and Technology (IJCET), Vlo.4, No.3, May-June, 2013.

[20] Stan Z. Li and Anil K. Jain, "**Handbook of Face Recognition**", ISBN: 0-387-40595-X, Springer Verlag, 2005.

[21] A. K. Jain, P. Flynn, and A. A. Ross, "**Handbook of Biometrics**", ISBN-13: 978-0-387-71040-2, Library of Congress Control Number: 2007934776, www.Springer.com, 2008.

[22] V. Roselin, L.M.Waghmare, and E.R.Chirchi, "**Iris Biometric Recognition for Person Identification in Security Systems**", International Journal of Computer Applications, Vol. 24, No. 9, 0975 – 8887, 2011.

[23] D. Rudrapal, S. Das and N. Kar, "**Voice Recognition and Authentication as a Proficient Biometric Tool and its Application in Online Exam for P.H People**", International Journal of Computer Applications, Vol. 39– No.12, 0975 – 8887, February 2012.

[24] S. Sharma, R. Tiwari, A. shukla, and V. Singh, "**Identification of People Using Gait Biometrics**", International Journal of Machine Learning and Computing, Vol. 1, No. 4, October, 2011.

[25] F. Monrose and A. Rubin, "**Authentication Via Keystroke Dynamics**", In Proceedings of Fourth ACM Conference on Computer and Communications Security, pages 48–56, Zurich, Switzerland, April 1997.

[26] S. P. Banerjee, and D. L. Woodard, "**Biometric Authentication and Identification using Keystroke Dynamics**", Journal of Pattern Recognition Research 7(2012), 116-139, July, 2012.

[27] D. Maltoni, D. Maio, A. K. Jain and S. Prabhakar, "**Handbook of Fingerprint Recognition**", ISBN: 978-1-84882-253-5, Springer Verlag, London, 2009.

[28] L.I.Burke, "Introduction to artificial neural systems for pattern recognition", Computers & Operations Research, Vol. 18, No. 2, pp. 211-220, 1991.

[29] R. Cappelli, A. Lumini, D. Maio and D.Maltoni, "**Fingerprint Classification by Directional Image Partitioning**", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 5, May 1999.

[30] A.K. Jain, S. Prabhakar, L. Hong and S. Pankanti, "**Filterbank-Based Fingerprint Matching**", IEEE Transaction on Image Processing, Vol. 9, No. 5, May 2000.

[31] B. M. Mehtre R and D. Division, "**Fingerprint Image Analysis for Automatic Identification**", Machine Vision and Applications, Springer Verlag, 1993.

[32] I. G. Babatunde, "**Fingerprint Matching Using Minutiae-Singular Points Network**", International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol. 8, No. 2, 2015.

[33] P. Deshmukh, S. Pathan and R. Pathan, "**Image Enhancement Techniques for Fingerprint Identification**", International Journal of Scientific and Research Publications, Vol. 3, No. 3, March 2013.

[34] S. Greenberg, M. Aladjem and D. Kogan, "**Fingerprint Image Enhancement using Filtering Techniques**", Real-Time Imaging 8, 227–236, Elsevier Science Ltd, 2002.

[35] L. O'Gonnan and J. V. Nickerson, "**Matched Filter Design for Fingerprint Image Enhancement**", International Conference on Acoustics, Speech, and Signal Processing, IEEE, Vol. 2, April 1988.

[36] B. G. Sherlock, D. M. Monro and K. Millard, "**Fingerprint Enhancement by Directional Fourier Filtering**", IEE Proc.-Vis. Image Signal Process, Vol.141, No.2, April 1994.

[37] A. Almansa and L. Cohen, "**Fingerprint image matching by minimization of a thin-plate energy using a two-step algorithm with auxiliary variables**", IEEE Workshop on Applications of Computer Vision (W ACV'00), 4-6 December 2000, Palm Springs, CA, USA.

[38] S. G. MALLAT, "**A Theory for Multiresolution Signal Decomposition: The Wavelet Representation**", IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 11, No. 7, JULY 1989

[39] A. Motwakel and A. Shaout, "**Fingerprint Image Enhancement and Quality Analysis – A Survey**", Article, International journal of Computer Networks and Communications Security, September 2014.

[40] F. Zhao and X. Tang, "**Preprocessing for Skeleton-Based Fingerprint Minutiae Extraction**", ELSEVIER, Pattern Recognitio, Vol. 40, Issue. 4. April 2007.

[41] M. Redhu and Balkishan, "**Fingerprint Recognition Using Minutiae Extractar**", International Journal of Engineering Research and Applications (IJERA), Vol. 3, No. 4, pp .2488-2497, August 2013.

[42] Md. SFerdous, M. Chowdhury, Md. Moniruzzaman and F. Chowdhury, "**Identity Federations: A New Perspective for Bangladesh**", International Conference on Informatics, Electronics & Vision, P. 219 – 224, IEEE, 2012.

[43] W. Stallings, "**Cryptography and Network Security Principles and Practices** ", Fifth edition, Prentice Hall, 2011.

[44] M. Kulkarni, A. Yadav, D. Shah, P. Bhandari and S. Mahapatra, "**Unique ID Management**", Int.J.Computer Technology & Applications, Vol. 3,-No.2.

[45] A. Patnaik and D. Gupta, "**Unique Identification System**", International Journal of Computer Applications, Vol. 7, No. 5, Semptember 2010.

[46] R. Bani-Hani, Y. Wahsheh and M. Al-Sarhan, "**Secure QR Code System**", IEEE, 10th International Conference on Innovations in Information Technology, 2014.

[47] A. Sankara Narayanan, "**QR Codes and Security Solutions**", International Journal of Computer Science and Telecommunications, Vol. 3 No. 7, July 2012.

[48] N. Bhargava, A. kumawat and R. Bhargava, "**Demonstration of Barcodes to QR Codes through Text Using Document Software**", International

Journal of Innovative Research in Science, Engineering and Technology, Vol. 3, No. 9, September 2014.

[49] P. Gupta and S. Kumar, "**A Comparative Analysis of SHA and MD5 A Comparative Analysis of SHA and MD5**", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5, No. 3, 2014.

[50] Parshotam, R. Cheema and A. Gulati, "**Improving the Secure Socket Layer by Modifying the RSA Algorithm**", International Journal of Computer Science, Engineering and Applications (IJCSEA), Vol. 2 No. 3, June 2012.

[51] A. Johny and Jayasudha J. S, "**Secure Socket Layer Implementations-A Review**", International Journal of Computer Science & Engineering Technology (IJCSET), Vol. 4 No. 2, Feb 2013.

[52] http://www.sourceafis.org/, last access on 2nd January 2015.

[53] David Houcque, "**Introduction to MATLAB forEngineering Students**", Northwestern University, August 2005.

[54] Karli Watson, Christian Nagel, Jacob Hammer Pedersen, Jon D. Reid,Morgan Skinner and Eric White, "**Beginning Microsoft Visual C# 2010**", Wiley, 2010.

[55] Joyce Cox and Joan Lambert, "**Step by Step Microsoft Access 2013**", Microsoft press, 2013.

[56] http://www.zktechnology.com/, last access on 25th January 2015.

Appendix A

All users NIDN

Table A-1: 1st to 23th users

User Name	NIDN	Birthdate
Ali Kadom	ed507237030ee067e95e5cf33c06bae5	26/2/1976
Mntaser Saleem	b73a482561c9e96fd1830986384037a2	20/5/1990
Abdualhameed Mondel	207f0b300f45592a2356285abba1addf	3/5/1976
Safa Kamal	cb5c361cb7754235ed5b4696191cb8d6	11/1/1990
Hussan Kamal	e932792e96d4bf45c002654f9977eadf	30/7/1991
Thukra Jabar	27f5a08612af02a5040399582461f760	4/11/1970
Hurria Kadom	b5f2fd428574f51cd7f5b8f26a767fce	14/1/7/1958
Rafel Saleem	da612dafcc144ca10334add17b6dba66	1/6/1988
Mohammed Saleem	7fdb315db63c076082974794c33d2286	18/6/1982
Sabah Faleh	3bd27e91ece2697d1bb10f43bf4711f5	7/7/1989
Mustafa Ali	922fca9f9f70f8f850f7f0cfb12dfc54	20/11/1985
Hussain Ali	e6c259b556292e3fcd75d25a01b1ba4b	19/4/1988
Baqer Abass	334d50f2652f06ef9af2086c8e623b12	24/6/1989
Foad Emad	5bf7ba0626ba7539beae130460f63b49	2/3/1992
Haider Abass	4f225e3bf47bd3fc28f5850ec2e5f270	19/1/1984
Hassan Hashim	0327d831954c59b7f47ba75d8a70248a	1/1/1970
Khalel Ibrahim	26ad55c633683befaad0546ff859b568	1/7/1950
Yasser Abass	c4e7c7e1145ae5adf4a0b9390385655e	1/6/1990
Mustafa Ala	0b0dfed736b135a5a9699bbaecb3aada	6/3/1994
Falah Saleem	522a76d7147b24a44436d1364acdec00	25/5/1985
Mariam Hamid	8c06fe47d674a94a96e01e808a7a2185	15/10/1988
Mahmood Khalel	c9c254cd376c47c818a00bf94421ef43	1/7/1953
Salam Ahmed	5a52c8436e7b7b5f6fee585f5f831e37	9/2/1990

Table A-2: 24th to 47th users.

User Name	NIDN	Birthdate
Jwan Hammed	32a493f42014447d4802d5e88ae4801d	7/12/1991
Asima Qassim	d2aa7d309392cb1241202f207a07cec9	7/10/1992
Salwa Marwan	5ff9a5dc2ff8f5cf9a88b1302b2dd09e	2/7/1991
Noor Quasy	4dbf5d507983d43d127879ef6d5d0c05	5/5/1988
Mohammed Wahab	3cc802ed357de171a57b0e4a5bb32c04	5/5/1989
MohammedAbedaljabar	9ee05e3ac4bb17ccc9682da1f6569070	16/10/1982
Samer Faleh	35bdf7e7ffdf077def3b8967d28ed2a9	3/5/1961
Mohaned Majed	d500181cca593b8fd07cf59241ab3f5b	23/5/1992
Dhia Yahia	8ab5ff9b75c9697b3abd9c8887c19bf1	15/10/1994
Mustafa khuder	8dfa65a95e1669d95dce990fda53d2fe	10/7/1994
Saif Ali	86938fafb9847a3c949ffe82024082ba	14/1/1994
Karar Maki	841230fdc026143dff279fed34a3b445	25/8/1993
Ahmed Suher	b237bc6d0aba61964e9069ad8740bb9d	24/10/1993
Dhurgham Khari	e957a6073d75aedb3bd5650632314793	10/8/1993
Saif Nafi	581991792d545b2bf23c3accb48ec9ea	3/6/1995
Mustafa Haji	5f0c73aa1caccd272cca3c4442842e5d	11/11/1987
Osama Hanosh	a9a0df005ba8fcbb9ab97c97dfd2f907	3/3/1988
Mokhaled Sakleem	ff7bcafb6eebc6ad89243bbdf5cf18df	7/4/1993
Mustafa Taher	e6d888f863f4d1d14fc56f92cf42fc92	10/4/1995
Dhia Raad	e5f0555cf991d544696808d8eacbad17	20/7/1993
Zia Raad	5934a3420ee240fff4c53001cd710ea1	21/7/1994
Hasanen Ibrahim	b261323fa6562babdaf42e624f77340f	1/1/1989
Mahdi Satar	131db019ce05547d094da16531b88dfe	4/3/1985
Ahmed Saleem	ddc0c43317a215668d9dab3b6faa6976	19/1/1992

Appendix B

ZK 4500 fingerprint scanner specification

Table B-1: Technical specifications

Specification type	Measuring
Resolution	500 DPI/256 gray
Sensing Area	15x18 mm
Image Aize	280x360 pixel
Interface	USB 1.1/2.0
Supports Operating System	500 DPI/256 gray
Operating Temperature	0˚-55˚C / 32˚-131˚F
Operating Humidity	20%-80%
Color	Black
USB Cable	150 cm
Weight	0.24 kg
Dimension (WxHxD)	53x80x60 mm

المستخلـص

وثوقية شبكات الحواسيب تعتبر واحدة من خدمات تأمين الشبكات المهمة. طرق الوثوقية التقليدية مثل (**token –based and knowledge –based**) تعاني مشاكل متعددة (**السرقة,النسيان, التخمين** ...الخ.) حيث ادت هذه المشاكل الى استخدام ما يعرف بـ **"الوثوقية البايومترية"**.

تعتبر بصمة الاصابع واحدة من الخصائص البايومترية الفيزيولوجية التي تمتلك الحد الكافي من القدرة على تمييز الاشخاص . بصمة الاصابع امتلكت استقطاب واسع في التطبيقات الامنية .

يقدم هذا البحث نظام وثوقية لشبكات الحواسيب بالاعتماد على بصمة الاصابع كعنصر من العناصر البايومترية مع بعض المعلومات الشخصية الاعتمادية الثابتة مثل الاسم وتاريخ الميلاد. هذا النظام المقترح يولد رقم هوية وطني مميز بواسطة دمج الخصائص الدقيقة لبصمات الاصابع مع المعلومات الشخصية وطبع الرقم الناتج بصورة باركود ثنائي الابعاد (**QR image**) لاستخدامها كبطاقة عند دخول النظام . اضافة لذالك تم استخدام تطابق البصمات للتحقيق من هوية المستخدم في التطبيقات التي تحتاج مستوي حماية عالي.

يوفر النظام المقترح نوعين من خدمات المصادقة: **النوع الاول**, خدمة المصادقة الاعتيادية لحماية التطبيقات العامة التي تحوي بيانات عامة مثل انظمة دفع اجور القوائم والمستحقات, هذه الخدمة تحتاج الرقم الوطني او بطاقة الباركود لولوج النظام. **النوع الثاني**, خدمة المصادقة القوية لحماية التطبيقات الخاصة التي تحوي بيانات حساسة مثل المعاملات المصرفية, هذه الخدمة تحتاج الرقم الوطني او بطاقة الباركود مع بصمة الاصبع لنفس المستخدم لولوج النظام.

العمل التجريبي بين , مع استخدام قيمة عتبة التمييز دون ال(٥٠) , دقة قبول المستخدمين تكون (١٠٠%) , ومع قيمة عتبة التمييز تساوي ال (٥٠) , دقة قبول المستخدمين تكون (٩٦.١٥٣). زيادة قيمة عتبة التمييز تؤدي الى زيادة نسبة رفض المستخدمين عن طريق اسقاط قيم التطابق القليلة وابقاء القيم العالية. نقترح استخدام قيمة عتبة اكبر من ال (٥٠) للتطبيقات ذات المستوى الامني العالي.

وثوقية شبكة امنة بأعتماد الخصائص البايومترية في الرقم الوطني الموحد

رسالة

مقدمة الى كلية هندسة المعلومات في جامعة النهرين

وهي جزء من متطلبات نيل درجة ماجستير علوم في

هندسة الشبكات وتقنيات الشبكة الدولية

من قبل

منتصر سليم فالح

(بكالوريوس علوم في هندسة المعلومات والاتصالات ٢٠١٣ م)

ربيع الثاني ١٤٣٧ هـ

شباط 2016 م