**Sudan University of Science and Technology**

**Collage of Graduate Studies**

# FINGERPRINT IMAGE QUALITY ANALYSIS and ENHANCEMENT USING FUZZY LOGIC TECHNIQUE

**تحليل جودة صورة البصمة وتحسينها باستخدام تقنية المنطق الضبابي**

A Dissertation Submitted to the College of Graduate Studies,
Faculty of Computer Science and information Technology,
Sudan University of Science and Technology
In Partial Fulfillment of the
Requirements for the degree of
**DOCTOR OF PHILOSOPHY**
Major Subject: Fuzzy Image Processing

**by**

Abdelwahed Motwakel Eltayeb Ismaeil, M.Eng

**Supervisor**

Prof. Dr. Adnan K. Shaout

**January 2017**

بسم الله الرحمن الرحيم

قَالَ تَعَالَىٰ:

﴿ يَرْفَعِ ٱللَّهُ ٱلَّذِينَ ءَامَنُواْ مِنكُمْ وَٱلَّذِينَ أُوتُواْ ٱلْعِلْمَ دَرَجَٰتٍ ۚ وَٱللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ ﴿١١﴾ ﴾

﴿ أَيَحْسَبُ ٱلْإِنسَٰنُ أَلَّن نَّجْمَعَ عِظَامَهُۥ ﴿٣﴾ بَلَىٰ قَٰدِرِينَ عَلَىٰٓ أَن نُّسَوِّيَ بَنَانَهُۥ ﴿٤﴾ ﴾

المجادلة: ١١

القيامة: ٣ ـ ٤

## Dedication

*To the soul of my parents:*

*Motwakel Altayeb Ismaeil&Shal Muna Ali Khalifa.*

# ACKNOWLEDGEMENT

First and foremost I would like to express my sincerest gratitude to my supervisor, Professor Adnan Shaout, for his immense help, guidance, stimulating suggestions and encouragement all the time, andfor his supporting me throughout my dissertation with his patience and knowledge. He always provide a motivating and enthusiastic atmosphere to work with. I am grateful for the time that he spent discussing ideas, revising papers and helping me work on this research. It was a great pleasure to do this dissertation under his supervision.

I am thankful to Professor Izzeldin Mohamed Osman, for the motivation and inspiration that triggered me for the dissertation work. I would also like to thank Dr. Mohammed Al Hafiz Musa, Dr Hoida Ali and all the staff members of Computer Science and Information Technology College,who were always there at the need of the hour and provided with all the help and facilities, which I required for the completion of the dissertation.

Last but not the least, I express my heartfelt like to thank to all of my friends and colleagues who have encouraged, supported and providing me useful information during my work, Omer Balula Ali, Mohamed Khalid, andMalak Osman.

Finally, but most importantly, I express my heartfelt like to thank my family; my wife Om Hani, and my daughters Muna, Rawan and Saharwho stood by my side and for their understanding, prayers and endless support.

**ABSTRACT (English)**

The quality of fingerprint image greatly affects the performance of minutiae extraction and the process of matching in fingerprint identification system. In order to improve the performance of the fingerprint identification system, a fuzzy techniques used for bothfingerprint image quality analysis and enhancement.

First, the quality analysis performed by extracting four features from a fingerprint image which are the local clarity score (LCS), global clarity score (GCS), ridge_valley thickness ratio (RVTR), and the Global Contrast Factor (GCF). Afuzzy logic technique that uses Mamdani fuzzy rule model was designed. The fuzzy inference system was able toanalyze and determinate the fingerprint image type (oily, dry or neutral) based on the extracted feature values and fuzzy inference rules.

The experimental result obtained using the fuzzy technique were successful, and was able to determine the fingerprint image quality (Oily, Neutral, or Dry) according to their input features.

Secondly, fuzzy morphology was applied to enhance dry and oily fingerprint images.Fuzzy morphology method improves the quality of a fingerprint image, thus, improving the performance of the fingerprint identification system significantly.

All the experimental work which was done for both quality analysis and image enhancementwas done using the DB_ITS_2009 database which is a private database collected by the department of electrical engineering, institute of technology SepuluhNopember Surabaya, Indonesia.

The performance evaluation was done by using Feature Similarity index (FSIM). Where the FSIMis an image quality assessment (IQA) metric, whichuses computational models to measure the image quality consistently with subjective evaluations.

## ABSTRACT (Arabic)
## المستخلص


جودة صورة بصمة الأصبع تؤثر بشكل كبير على أداء عملية استخراج مكونات بصمة الأصبع وبالتالي تؤثر على عملية المطابقة في نظام التعرف على بصمات الأصابع. من أجل تحسين أداء نظام التعرف على بصمات الأصابع، تم استخدام تقنيات المنطق الضبابي لتحليل جودة صورة بصمة الأصبع وتحسينها.

أولا، تحليل جودة صورة بصمة الأصبع وذلك باستخلاص أربعة سمات أساسية من صورة بصمة الأصبع و هي درجة الوضوح المحلية ، درجة الوضوح العامة ، ونسبة السماكة بين التلال والوديان ، وعامل التباين العام. تم تصميم تقنية المنطق الضبابي الذي يستخدم نموذج ممداني و نظام الاستدلال الضبابي لتحليل جودة الصورة ، هذه التقنية قادرة على تحليل جودة الصورة وتحديد نوع جودة الصورة إما الدهنية، أو الجافة أو محايدة. و ذلك استنادا إلى السمات الميزة التي تم استخراجها من صورة بصمة الأصبع وقواعد الاستدلال في المنطق الضبابي.

وكانت النتيجة التجريبية التي تم الحصول عليها باستخدام تقنية المنطق الضبابي ناجحة، وتمكنت من تحديد نوعية صورة بصمة الأصبع إما الدهنية، أو الجافة أو محايدة. بناءاً على السمات التي تم إدخالها.

ثانيا، تم تطبيق تقنية المنطق الضبابي التي تتعامل مع التشكل (علم المورفولوجيا) لتحسين جودة صورة بصمة الأصبع الجافة والدهنية. باستخدام هذه التقنية تم تحسين جودة صورة بصمة الأصبع الجافة والدهنية بشكل ملحوظ، وبالتالي تم تحسين أداء نظام التعرف على بصمات الأصابع.

وقد تم عمل كل التجارب لكل من تحليل نوعية جودة الصورة وتحسين الصورة الجافة والدهنية باستخدام قاعدة بيانات تسمي DB_ITS_2009 و هي قاعدة بيانات خاصة تم تجميعها من قبل قسم الهندسة الكهربائية، معهد العاشر من نوفمبر للتكنولوجيا بمدينة سورابايا .

وقد تم تقييم أداء الطريقة باستخدام مقياس السمات المتشابهة هو احد مقاييس تقييم جودة الصور والذي يستخدم نماذج حسابية لقياس جودة الصورة مع بعض التقييم الذاتي . حيث مؤشر مقياس السمات المتشابهة أشار إلى تحسن ملحوظ في كل من صورة بصمة الأصبع الجافة والدهنية.

## Table of Contents

## List of Tables

## List of Figures

## LIST OF SYMBOLS / ABBREVIATIONS

| | |
|---|---|
| GCS | Global Clarity Score |
| RVTR | Ridge Valley Thickness Ratio |
| GCF | Global Contrast Factor |
| FIS | Fuzzy inference system |
| FIE | Fingerprint image enhancement |
| LCS | Local Clarity Score |
| AFIS | automatic fingerprint identification system |
| FL | Fuzzy Logic |
| FIP | Fuzzy Image Processing |
| SE | structure element |
| FSIM | The feature-similarity |
| DNA | Deoxyribo Nucleic Acid |
| HVS | human visual system |
| PC | phase congruency |
| GM | Gradient magnitude |
| DFT | Discrete Fourier Transform |
| IDFT | Inverse Discrete Fourier Transform |

# CHAPTER I

# Introduction

## 1.1. Background

Humans identify others based on the characteristics and features which are used to distinguish human being such as facial features or voice which are used to recognize a person. These characteristics and features of human being, called biometrics, are used daily to identify each other.

Biometrics is a scientific term refers to the use of the unique physiological and behaviors characteristics of the human body to distinguish each other [1]. There are many types of physiological and behaviors characteristics that are used in most security application to verify human being such as fingerprint, hand geometry, iris, retina, and face. Currently, fingerprint is the most commonly biometrics for security applications.

Fingerprint identification is the most widely used biometrics technology which is used in criminal investigations, commercial applications, and so on. With such a wide variety of uses for the technology, the demographics and environment conditions that it is used in are just as diverse. However, the identification performance of such system is very sensitive to the quality of the captured fingerprint image. Fingerprint image quality analysis or assessment is useful in improving the performance of fingerprint identification systems.

In many systems it is preferable to substitute low quality images for better ones. Therefore, image quality analysis takes an important part of image processing. The quantification of image quality allows to tune a system and to accurately evaluate an input image performance measurement [2].

Various factors can affect the quality of fingerprint images such as dryness/wetness conditions, non-uniform and inconsistent contact, permanent cuts and so on. Many of these factors cannot be avoided. Therefore, assessing the quality and validity of the captured fingerprint image is necessary and meaningful. Many papers in biometric literature address the problem of assessing fingerprint image quality. But these methods still have some problems and can’t be suitable for all conditions [2].

## 1.2. Image Enhancement

Fingerprint image enhancement is an essential preprocessing step in fingerprint recognition applications. The Image enhancement is a preprocessing technique performed to make the image clearer than the original image. Enhancement methods are needed since the fingerprint images acquired from sensors or other media are not always assured with perfect quality. In fingerprint recognition system, the enhancement methods are needed to increase the contrast between ridges and valleys and for connecting the false broken points of ridges due to insufficient amount of ink. Enhancement of fingerprint images is an essential step in any practical fingerprint based authentication system as fingerprint images that are acquired in real world are often noisy. In practice two humans with the same fingerprint have never been found, even the fingerprints in twins [3].

A fingerprint contains narrow ridges separated by narrow valleys and these ridges flows almost parallel to each other. The ridges are the dark areas of the fingerprint and the valleys are the white areas that exists between the ridges. The noises and degradations arise due to poor skin conditions, varying finger pressure while acquisition, sensor noise and dry, wet or oily fingers. Fingerprints may be degraded and corrupted with elements of noise due to many factors including variations in skin and impression conditions. This degradation can result ina significant number of spurious minutiae being created and genuine minutiae being ignored. High quality fingerprint image is very important for fingerprint verification to work properly [3].

Image enhancement techniques can be divided into three board categories; spatial domain methods, frequency domain methods, and fuzzy domain methods. Enhancement techniques use various combinations of methods from these three categories. A spatial domain method refers to the image plane itself, and image processing methods in this category are based on direct manipulation of pixels in an image. A frequency domain method consists of modifying the Fourier transform of an image and then computing the inverse transform, Discrete Fourier Transform (DFT), to get back to input image [2]. A fuzzy domain method has three main stages: image fuzzification, modification of membership values and image defuzzification. The fuzzification and defuzzification steps are due to the fact that we do not possess fuzzy hardware. Therefore, the coding of image

data (fuzzification)and decoding of the results (defuzzification) are steps that make possible to process images with fuzzy techniques. The main power of fuzzy image processing is in the middle step (modification of membership values) after the image data is transformed from gray-level plane to membership plane (fuzzification). An appropriate fuzzy technique to modify the membership values is needed which can be a fuzzy clustering or a fuzzy rule-based approach or a fuzzy integration approach. Fuzzy logic offer us powerful tools to represent and process human knowledge in the form of fuzzy if-then rules [4].

### 1.3. Fingerprint Image Quality Analysis

Image quality assessment is one of the key technologies in various image processing applications. Fingerprint image quality assessment is important in eliminating poor fingerprint images, which will also affect the performance of the automatic fingerprint identification system. However, a lot of factors can affect the quality of fingerprint images, dryness/oiliness conditions, non-uniform and inconsistent contact, permanent cuts and so on. Many of these factors cannot be avoided. Therefore, assessing the quality and validity of the captured fingerprint image is necessary and meaningful [5].

In general, the fingerprint image quality relies on the clearness of separated ridges from valleys and the uniformity of the separation. A fingerprint image changes in many ways because of the change in environmental conditions such as temperature, humidity and pressure. The overall quality of the fingerprint depends greatly on the condition of the skin. Dry skin tends to cause inconsistent contact of the finger ridges with the scanner's platen surface, causing broken ridges and many white pixels replacing ridge structure. To the contrary, the valleys on the oily skin tend to fill up with moisture, causing them to appear black in the image similar to ridge structure [6].

### 1.4. Statement of The Problem

The performance of fingerprint image matching algorithms heavily depends upon the quality of the input fingerprint images. It is very important to acquire good quality images, however, in practice a significant percentage of acquired images are of poor quality due to environment sensor factors or to the finger state of the user. The ridge structure in a poor quality image may not be well defined and hence may not be detected correctly. A poor fingerprint image matching suffer from many problems such as the following:

a) A significant number of spurious minutiae pattern are created.

b) A large percentage of genuine minutiae is ignored.
c) A false matching can be detected.

Therefore, an efficient and an intelligent image enhancement algorithm is needed to increase the performance of minutiae matching algorithm. To achieve that, a fingerprint image quality assessment algorithm is required which can measure the fingerprint quality images. The type of images the proposed algorithm will assess are as follows:

a) Oily images, where some parts of valleys may be filled up causing them to appear dark or adjacent ridges appearing stand close to each other in many regions. This type of fingerprint images make ridges look very thick.
b) Neutral images, where images are not very effective and have uniform separation between ridges and valleys.
c) Dry images, where the ridges appear to be broken and having many white pixels in the ridges.

### 1.5. Research Significance:

Nowadays with the advancements in technology, people are getting informed and educated through the latest communication technology. As a result, it has become easy to forge data for the purpose of vandalism so information and data must be protected. Fingerprint recognition system may be used to verify the identity and to ensure the privacy and protection of confidential information.

The fingerprint recognition system is considered to be one of the most effective and secure systems in information security because it cannot be forgotten or forged as it depends on the physiological characteristics of the human being.

We need to improve fingerprint image quality to recover the original image and to increase the performance of fingerprint recognition system due to some corrupted, noise, skin variations and impression conditions, or changes in environmental conditions such as temperature, humidity and pressure.

## 1.6. Research Questions

The main questions that will be addressed in this research is how to use fuzzy logic techniques to assess and enhance fingerprint images? There are additional sub-questions such as the following:

- What is the method and technology that will be used for dry fingerprint image enhancement and is it the most suitable method and technology?
- What is the method and technology that will be used for oily fingerprint image enhancement and is it suitable method and technology?
- What is the method and technology that will be used for assessing fingerprint image quality and is it a suitable method and technology and what are the features that will be extracted for this process, and how good is the assessment the fingerprint image?
- What are the databases that will be used for assessment and enhancement?

## 1.7. Research Hypothesis

It is hypothesized that fingerprint image quality will analyze and determine three types; oily, dry and neutral based on fingerprint features using fuzzy logic. It is also hypothesized that oily and dry fingerprint image will be enhanced based on fuzzy logic techniques and morphological operations.

## 1.8. Research Philosophy

Fingerprint recognition system with good fingerprint image quality could introduce many benefits such as the decline of crimes and reducing the losses of wealth and assets. There are many economic benefits that people could gain in using a fingerprint system. For example, a fingerprint recognition system that is used in taking the attendance of employees at work can insure that workers are attending or not and therefore the overall performance may be positively affected.

On the other hand, a fingerprint recognition system based on good quality fingerprint images can identify criminals and keep sufficient information about them, which allow community to track criminals making the community safer.

## 1.9. Objective of The Research

The main objective of this research is to develop intelligent algorithms to analyze and enhancement fingerprint images with enough accuracy. Other objectives are as follows:

- To design a technique using fuzzy logic to analyze fingerprint image quality.
- To design a technique using fuzzy logic to improve dry fingerprint image quality.
- To design a technique using fuzzy logic to improve oily fingerprint image quality.

## 1.10. The Scope

Scope of this research is stated as follows:

- Analysis fingerprint image quality using a fuzzy inference system (FIS).
- Enhancement of dry fingerprint image using fuzzy image processing with morphological operations.
- Enhancement of oily fingerprint image using fuzzy image processing with morphological operations.
- The system should not enhancement fingerprint image if they are not oily nor dry.
- The system will not determine the fingerprint image quality if not oily nor dry.

## 1.11. Organization of the dissertation

This dissertation is organized as follows:

Chapter 2 gives an overview of the background materials used throughout the dissertation such as biometric system, fuzzy logic, fuzzy image processing, and technologies such fingerprint identification system. And the fingerprint image quality metrics.

Chapter 3 reviews the main existing methods in the research area referring to the fingerprint image quality analysis and enhancement. In particular, three categories of methods are outlined in image enhancement: (i) Spatial domain methods (ii) Frequency domain methods (iii) Fuzzy domain methods. Also reviews the open issues in research area.

Chapter 4 describes the methodology used to analyze fingerprint image quality as well as image enhancement.

Chapter 5describes implementation of the fingerprint image quality analysis by using fuzzy inference system and examines the performance of the methodology by using features similarity index metrics.

Chapter 6 describes implementation of the dry fingerprint image enhancement by using fuzzy morphology and examines the performance of the methodology by using features similarity index metrics.

Chapter 7 describes implementation of the oily fingerprint image enhancement by using fuzzy morphology and examines the performance of the methodology by using features similarity index metrics.

Chapter 8gives a brief conclusion and discussion about the methodology and its performance, and future work.

# CHAPTER II
# Background Material

Here we introduce the background material for the dissertation and we talk about specific techniques and approaches that we used in this research.

## 2.1. Introduction

In this chapter, the background material required to develop the work undertaken in this dissertation is discussed. In Section 2.2, Biometrics systems are described followed by types of biometrics and fingerprint identification. In Section 2.3, fuzzy logic are discussed, followed by the fuzzy image processing, fuzzy sets theory, Mathematical morphology and morphological operations. In Section 2.4. Performance metrics used to measure the system performance is discussed.

## 2.2. Biometric System

Fingerprint identification is one of most reliable methods for identifying individuals since there are no two persons having the same fingerprint. A fingerprint also remains unchanged over the lifetime of a person and is easy to acquire. Because of the large amount of fingerprints and recent advances in computer technology, there has been increasing interest in automatic fingerprint identification system (AFIS). The performance of AFIS depends heavily on the quality of a fingerprint image which is affected by many physical factors, such as dirty finger, dirty sensor face, etc. thus, measuring the quality of a fingerprint image as well as enhancing it becomes an important factor in AFIS [7].

Biometrics is the term used to recognize or identify a human being by using physiological and/or behavioral characteristics of a person. Biometrics are currently widely used in information security.

Biometric recognition system refers to the use of physiological (e.g., fingerprints, face, retina, iris) and behavioral (e.g., gait, signature) characteristics, called Biometrics for automatically recognizing individuals. Biometric recognition system works by using equipment to match information which is extracted from an individual with biometric information stored in the system. Physiological Biometrics are based on data extracted

from part of a human body which it unique to each person even twins, while behavioral Biometrics are based on action taken by a person such as keystroke which also is unique to each person[8].

In practice, the biometric system is a mixture of physiological and behavioral properties which cannot be a fully separated. For example, a fingerprint is physiological but when using an input device to acquire data from a finger, it depends on the person's behavior entrance. Thus, the inputs to the recognition system are a combination of physiological and behavioral characteristics [9].

A biometric system is essentially a pattern recognition system that operates by extracting features from part of a human body (called data set) which are unique to each person, and comparing the extracted features with stored features in database. According to the application context, the biometric system can work either in verification mode or in identification mode [1].

In the verification mode, the system authenticates a person's identity by comparing the captured biometric characteristic (data) with previously captured (enrolled) biometric reference template(s) pre-stored in the system [1]. The verification is done to decide if the identity by an individual is true. Verification system is applied to prevent multiple people of using the same identity, thus a verification system either rejects or accepts the submitted claim of identity.

In the identification mode, the system authenticates an individual by searching the entire enrollment templates of all the users in the database for a match. Therefore, the system performs a one-to-many comparison to establish if the individual's identity is present in the database and if so, returns the identifier of the enrollment reference that match. If the subject is not enrolled in the system database, then the identification is a fail. The purpose of the identification system is to prevent a single person from using multiple identities [1]. Identification may also be used in positive recognition for convenience, (e.g. the user withdrawal money from ATM without ATM card or PIN) better security and high efficiency. A biometric system can be applied in many application technologies such as forensic (criminal investigation), civilian (national ID, passport control, driver's license) and commercial (ATM, access control, e-commerce).

### 2.2.1. Types of Biometrics

There are many human characteristics that distinguish a human being from others. Fingerprints, for example, are the most popular biometrics technologies and have been used for over one hundred years and, therefore, are generally well accepted as a recognition technology. Other technologies such as face, hand geometry, speaker and iris recognition are also accepted. A biometric that would require giving a blood sample for frequent personal verification would probably not be well accepted [1]. The essential features that biometrics use from individuals are unique and these characteristics can't be copied or changed or swapped from one person to another. A brief introduction to the commonly used biometrics are given as follows:

**DNA**: Deoxyribo Nucleic Acid (DNA) is the part of a cell that contains genetic information which is unique for each individual, except for identical twins. It is, however, currently used widely in the context of forensic applications for person recognition [1].There are issues limiting the use of DNA biometrics in applications. Some of these issues are as follows:

(i) Contamination and sensitivity

(ii) Automatic real-time recognition issues

(iii) Privacy issues

**Ear:** Is the organ that detects sound. Not only receives sound, but also aids in balance and body position. Ear is part of the auditory system and it has been suggested that the shape of the ear and the structure of the cartilaginous tissue of the pinna are distinctive. The ear recognition approaches are based on matching the distance of salient points on the pinna from a landmark location on the ear. The features of an ear are not expected to be very distinctive in establishing the identity of an individual [1].

**Face**: Facial images are probably the most common biometric characteristic used by humans to make a personal recognition. The applications of facial recognition ranges from a static, controlled "mug-shot" verification to a dynamic, uncontrolled face identification in a cluttered background (e.g., airport). The most popular approaches to face recognition are based on either (i) the location and shape of facial attributes, such as the eyes, eyebrows, nose, lips, and chin and their spatial relationships, or (ii) the overall (global) analysis of the

face image that represents a face as a weighted combination of a number of canonical faces [1].

**Fingerprints**: The recognition of fingerprints is the oldest used biometrics system which goes back to more than hundred years. Fingerprint is the mark (ridge end and ridge bifurcation) left behind when a person touches an object with their finger. Fingerprints are currently used in most biometric system technologies and are becoming more mature. Most identification applications are based on biometric systems using fingerprint recognition. By 1998, fingerprint recognition products accounted for 78% of the total sales of biometric technologies. Fingerprint recognition products for desktop and laptop access are widely available from many different vendors at a low cost. This biometric technology uses the unique characteristics to each person, even identical twins, which it called ridges and valleys. A ridge is defined as a single curved segment and a valley is defined as the region between two adjacent ridges. One of the challenges of fingerprint technology is individuals that have poorly defined (tenuous) ridges in their fingerprints [10].

**Iris**: Every human being has an iris which is different from others, even identical twins. Therefore it is used to uniquely identify individuals. Iris is formed in the womb of women between the third and the eighth month during the formation of the fetus just like the fingerprint patterns. So the iris characteristics in each person are different even for the two eyes of a single individual, and they appear to be stable throughout life [8].

**Signature Verification**: Signature is a collection of ambiguous lines, or words or images or graphics or any other form which reflects the identity used in official business that carries the identity of the signed person. Signature verification has unique characteristics (speed, pressure and angle) that can distinguish each person. One focus for this technology has been the e-business applications [10].

**Voice Recognition**: The voice is automatic frequency or wave which is able to move among several material types such as solids, liquids, and gases, and does not spread in a vacuum. Voice has several characteristics such as size and shape of the throat and mouth which are unique for each individual. Voice recognition or speaker recognition is the problem of identifying a speaker from a short speech. This biometric technology uses the voiced features of speech that have been found to differ among individuals. Voice recognition has several difficulties such as background noise. Speaker recognition is most

appropriate in phone-based applications but the voice signal over phone is typically degraded in quality by the microphone and the communication channel [8].

**Gait**: Is the peculiar way one walks and is a complex spatio-temporal biometric. Gait is not supposed to be very distinctive, but is sufficiently discriminatory to allow verification in some low-security applications. Gait is a behavioral biometric and may not remain invariant, especially over a long period of time, due to fluctuations in body weight, major injuries involving joints or brain, or due to inebriety [1].

**Keystroke:** It is hypothesized that each person types on a keyboard in an individual way. This behavioral biometric is not expected to be unique to each individual but it offers sufficient discriminatory information to permit identity verification. Keystroke dynamics is a behavioral biometric; for some individuals, one may expect to observe large variations in typical typing patterns [1].

**Odor**: It is known that each object exudes an odor that is characteristic of its chemical composition and this could be used for distinguishing various objects. A whiff of air surrounding an object is blown over an array of chemical sensors, each sensitive to a certain group of (aromatic) compounds. A component of the odor emitted by a human (or any animal) body is distinctive to a particular individual. It is not clear if the invariance in the body odor could be detected despite deodorant smells, and varying chemical composition of the surrounding environment [1].

**Palm-print**: The palms of the human hands contain pattern of ridges and valleys much like the fingerprints. The area of the palm is much larger than the area of a finger and as a result, palm-prints are expected to be even more distinctive than the fingerprints. Since palm-prints canners need to capture a large area, they are bulkier and more expensive than the fingerprint sensors. Human palms also contain additional distinctive features such as principal lines and wrinkles that can be captured even with a lower resolution scanner [1].

**Retinal**: The retinal vasculature is rich in structure and is supposed to be a characteristic of each individual and each eye. It is claimed to be the most secure biometric since it is not easy to change or replicate the retinal vasculature [1].

### 2.2.2. Fingerprint Identification

Fingerprint identification refers to the process of potentially recognizing a certain fingerprint within a large database of fingerprints. The reader should be aware of the fact

that this technical definition of identification may lead to confusion compared to its usual interpretation in establishing the identity of a person. For the purpose of massive fingerprint identification, automated systems based on special hardware and software have been developed and are called Automated Fingerprint Identification Systems (AFIS) [11]. Figure 2.1 depicts the automated process in more detail. Its main elements are as follows:

- Acquisition of fingerprint image (by an acquisition device)
- Image processing (to enhance the relevant information)
- Feature extraction (to encode the found information)
- Storing and comparison of fingerprints

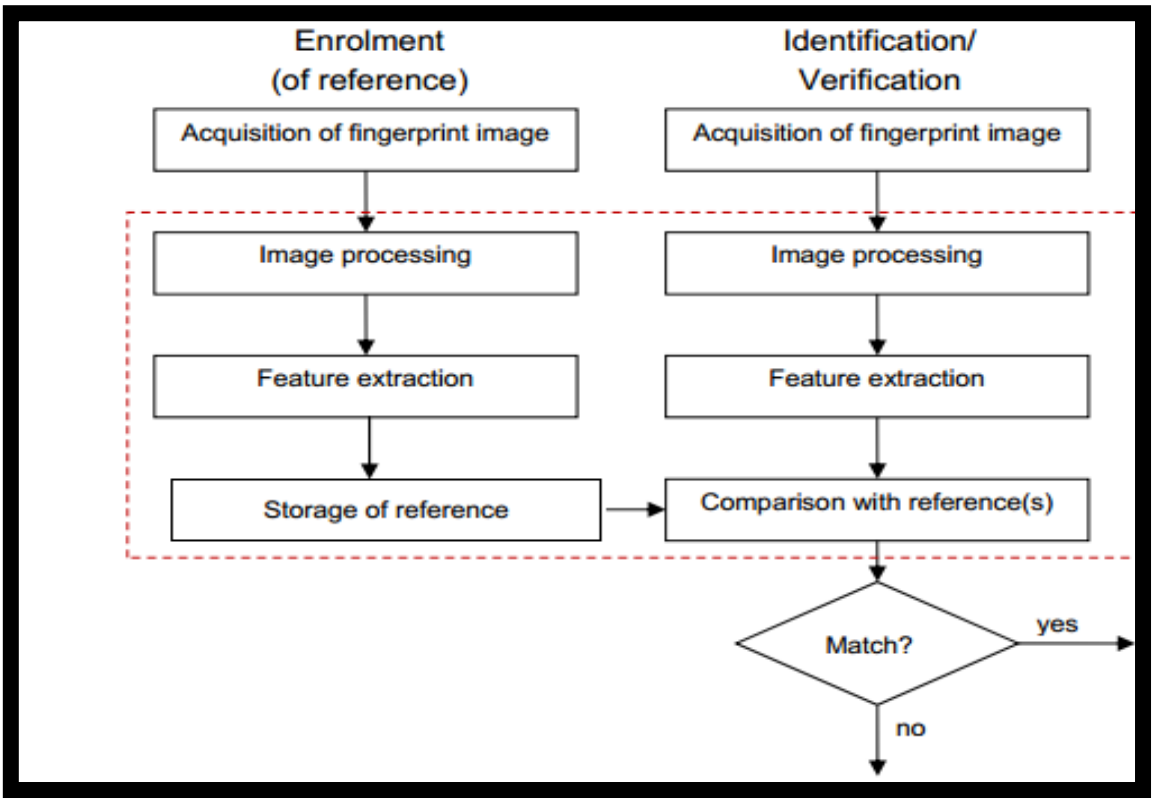


Figure 2. 1. Fingerprint Identification Process [11]

### 2.3. Fuzzy Logic

Fuzzy Logic was initiated in 1965 by Lotfi A. Zadeh, a professor for computer science at the University of California in Berkeley. Basically, Fuzzy Logic (FL) is a multivalued logic that allows intermediate values to be defined between conventional evaluations like true/false, yes/no, high/low, etc. Notions like rather tall or very fast can be formulated

mathematically and processed by computers in order to apply a more human-like way of programming computers [4].

Linguistic variables are used to express fuzzy rules, which can facilitate the construct of rule-based fuzzy systems. A linguistic variable can be defined as a variable whose values are words or sentences. For example, a linguistic variable such as age may have a value such as young, very young, old, and very old rather than 30, 36, or 18. However, the advantage of linguistic variables is that they can be changed via hedges (fuzzy unary operators) [12,4].

A fuzzy inference system is a method that interprets the values in the input vectors and based on user defined rules, assigns values to the output vector. Using a graphical user interface (GUI) editors and viewers in MATLAB fuzzy logic toolbox, we can build the rules set, define membership functions and analyze the behavior of a fuzzy inference system. The editors and viewers are used to edit and view the membership functions and rules for fuzzy inference system [4].

### 2.3.1 Fuzzy Image Processing (FIP)

Ability to represent the concept of partial truthiness, makes fuzzy suitable for image processing. Even if a solution for solving a particular problem is not clear, fuzzy logic can offer a solution using linguistic values for input(s) and output(s) variables. The main steps in designing fuzzy image processing system are defined as follows [4]:

1. Fuzzification
2. Inference Engine
3. Defuzzification

Fuzzification is nothing but assigning required membership function to map images from pixel plane to fuzzy plane. To fuzzify an 8x8 image of pixel values between 0 and 255, the fuzzy plane values would be values with the closed interval [0, 1] to represent the fuzziness of pixels. In the fuzzy plane, with the help of enhancement / transform operators, images can be modified. This plane is sometimes called inference plane. In the defuzzification plane, images from fuzzy plane are taken back to pixel plane with inverse transform. Fig. 2.2 shows the block diagram of FIP [4].

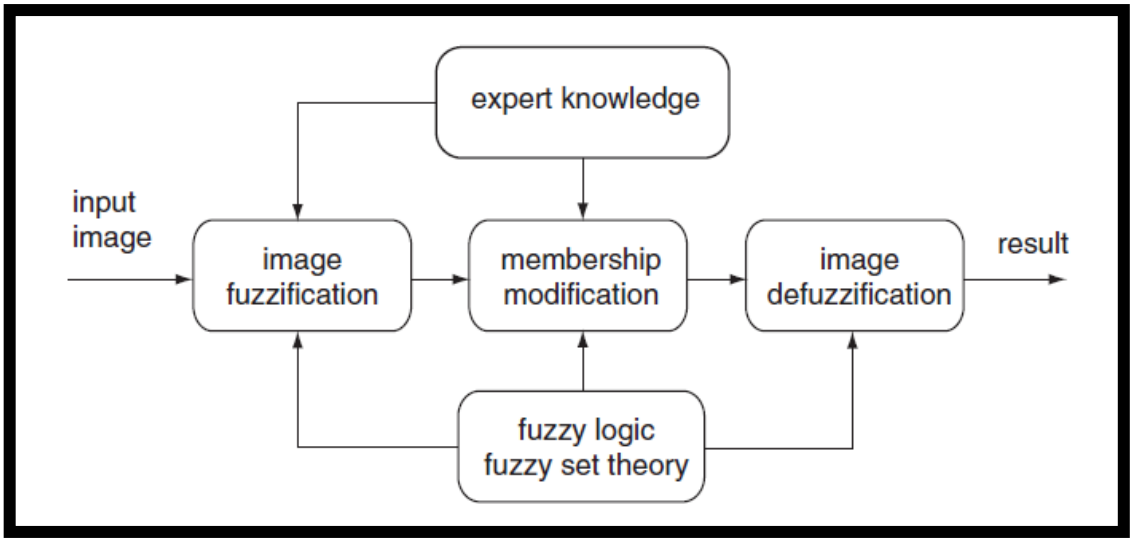


Figure 2. 2. General structure of fuzzy image processing.

Fuzzy image processing has three main stages: image fuzzification, modification of membership values, and, if necessary, image defuzzification. The fuzzification and defuzzification steps are due to the fact that we do not possess fuzzy hardware. Therefore, the coding of image data (fuzzification) and decoding of the results (defuzzification) are steps that make possible to process images with fuzzy techniques. The main power of fuzzy image processing is in the middle step (modification of membership values, see Figure2.3). After the image data are transformed from gray-level plane to the membership plane (fuzzification), appropriate fuzzy techniques modify the membership values. This can be a fuzzy clustering; a fuzzy rule-based approach, a fuzzy integration approach, etc.

### 2.3.2 Fuzzy Sets Theory

A set is a collection of objects called elements. The use of the word set means that there is also a method to determine whether or not a particular object belongs to the set. An object contained in a set is called a member or an element of the set. For example, it is easy to decide that the number 6 is in the set consisting of the integers 1 through 9. After all, there are nine objects to consider and it is clear that6 is one of them by simply checking all nine. In this section, capital letters denote sets, while members of a set are written in lower-case. To indicate the universe of discourse, often referred to as the universal set, we use U. All sets are members of the universal set. Additionally, a set with no elements is called a null, or empty set [13].

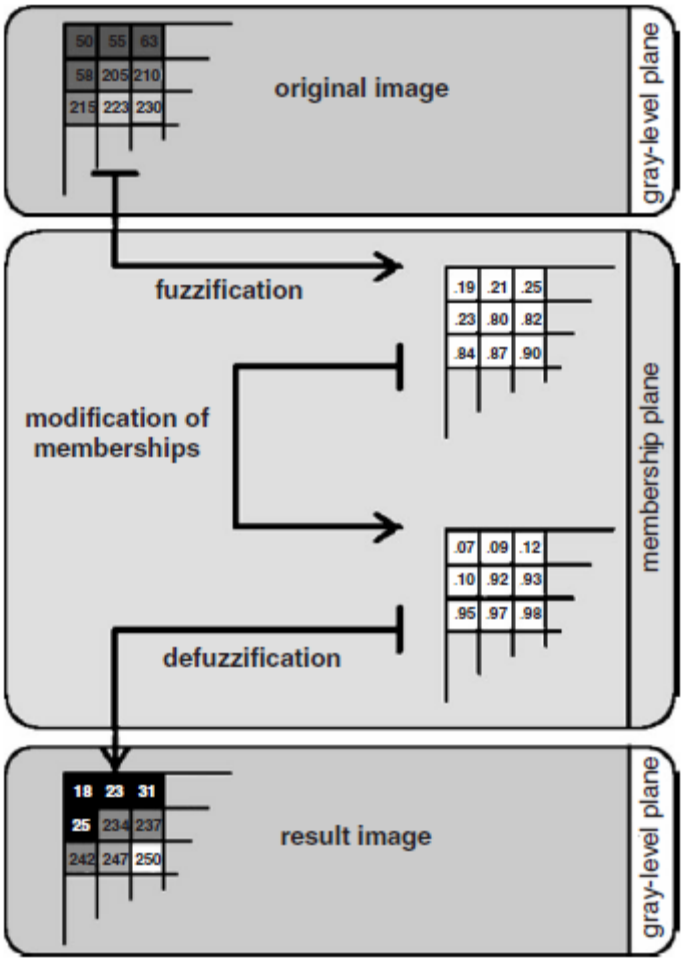


Figure 2. 3. Steps of fuzzy image processing [4]

If we have an element x of set A, it can be represented as follows:

$$x \in A$$

while if x is not a member of A, then it can be written as follows:

$$x \notin A$$

There are two methods used to describe the contents of a set; the list method and the rule method. The list method defines the members of a set by listing each object in the set A={a1,a2,...,an}.

The rule method defines the rules that each member must adhere to in order to be considered a member of the set. It is represented as follows:

A= {a| a has properties P1, P2, ...,Pn}.

When every element in the set A is also a member of set B, then A is a subset of B, which can be represented as follows:

$$A \subseteq B$$

If every element in A is also in B and every element in B is also in A, i.e. A⊆B and B⊆A, Then A and B are equal, which can be represented as follows:

$$A = B$$

If at least one element in A is not in B or at least one element in B is not in A, then *A* and B are not equal, which can be represented as follows:

$$A \neq B$$

Set A is a proper subset of B if A is a subset B but A and B are not equal, i.e. A⊆B and A≠B, which can be represented as follows:

$$A \subset B$$

To present the notion that an object is a member of a set either fully or not at all, we introduce the function μ. For every x∈U, $\mu_A(x)$ assigns a value that determines the strength of membership of each x in the set A,

$$\mu_A(x) = \begin{cases} 1 & if\ and\ only\ if\ x \in A, \\ 0 & if\ and\ only\ if\ x \notin A \end{cases}$$

Therefore, $\mu_A$ maps all elements of the universal set into the set A with values 0 and1, which is

$\mu_A$:U→ {0,1}. A here is an ordinary set. If $\mu_A$:U→ [0,1], then A is a fuzzy set.

### 2.3.3 Mathematical Morphology

For the purpose of image analysis and pattern recognition there is always a need to transform an image into another better represented form. During the past five decades image-processing techniques have been developed tremendously and mathematical morphology in particular has been continuously developing. Mathematical morphology is receiving a great deal of attention because it provides a quantitative description of geometric structure and shape and also it is a mathematical description of algebra, topology, probability, and integral geometry. Mathematical morphology is extremely useful in many image processing and analysis applications [13, 14].

Mathematical morphology denotes a branch of biology that deals with the forms and structures of animals and plants. It analyzes the shapes and forms of objects. In computer vision, it is used as a tool to extract image components that are useful in their presentation and description of object shape. It is mathematical in the sense that the analysis is based on set theory, topology, lattice algebra, function, etc. [13].

Mathematical Morphology is an approach based on set theory for extracting geometrical features out of signals. Morphological operations have been generally performed by using the structure element (SE). The most basic morphological operations are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels from object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the structuring element. While morphological operations usually are performed on binary images, some processing techniques also apply to grayscale images. Grayscale erosions and dilations produce results identical to the nonlinear minimum and maximum filters [14].

Mathematical Morphology comprises an important tool set for analyzing spatial structures in images. For binary images, the definitions of the fundamental morphological operations dilation and erosion can be related to the set-theoretic Minkowski addition and subtraction. The extension of those operations to grayscale images is strongly related to ranking operations and, therefore, to the concept of ordered sets. It has been considered, for a long time, to extend mathematical morphology to the case of fuzzy sets (as was done in other image processing disciplines).

The alpha-morphology approach is based on the level or alpha-set of a fuzzy membership degree function. Since each level set is a classical set by itself, standard morphological operations can be applied to them. After doing so for each level, the dilated or eroded level sets can be reunified to a grayscale image. Bloch et al. provided a formula for simplifying this computation [13].

However, by investigating the use of triangular norms in this simplification, Bloch et al. discovered a further family of fuzzy morphologies. For the first time, such operations have a formal degree of freedom, by allowing for replacing the occurrence of a minimum operation within the analytical expression of the operation with either formal expression of

a triangular norm. Also, Sinha and Dougherty in their approach considered the interpretation of the inclusion [13].

There are several approaches of fuzzy morphology but each approach of a fuzzy morphology gives a certain aspect of the underlying reasoning which has to be fuzzified. This could be the image data (grayscale images), the structuring element, or the set inclusion involved in the definition of Minkowski subtraction. We also propose the fuzzification of the pixel neighborhood, which leads to a further formal approach to fuzzy morphology using triangular norms and co-norms. Figure 2.4 shows the fuzzy morphologies, as they evolved from each other [13, 14].

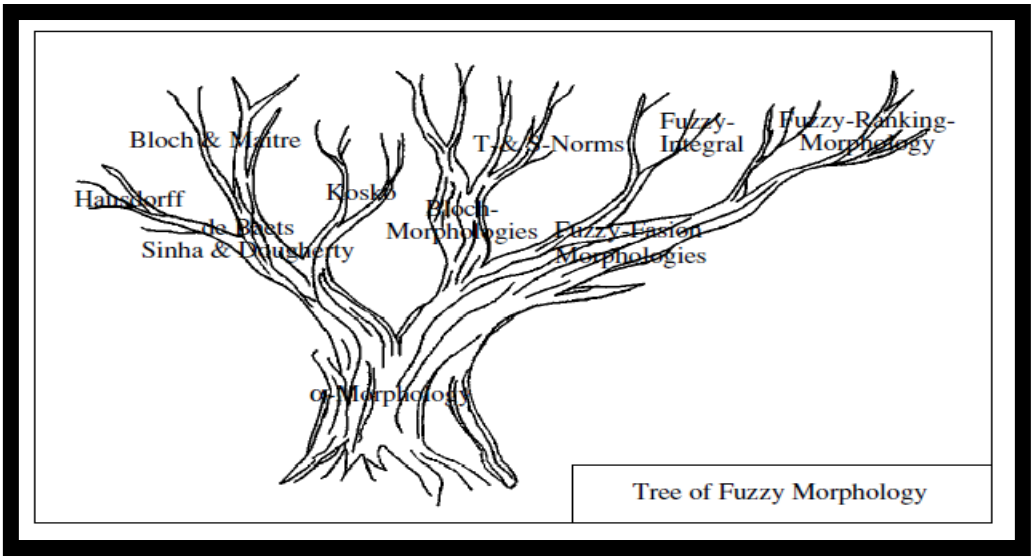


Figure 2. 4. The lineage of fuzzy morphologies [14].

### 2.3.4 **Morphological operations**

The two principal morphological operations are dilation and erosion. Dilation allows objects to expand, thus potentially filling in small holes and connecting disjoint objects. Erosion shrink objects by etching away (eroding) their boundaries. These operations can be customized for an application by the proper selection of the structuring element, which determines exactly how the objects will be dilated or eroded [15].

### 2.3.4.1 **The dilation**

The dilation process is performed by laying the structuring element B on an image A and sliding it across the image in a manner similar to convolution. The difference is in the operation performed. It is best described in a sequence of steps as follows [15]:

1. If the origin of the structuring element coincides with a 'white' pixel in the image, there is no change; move to the next pixel.

2. If the origin of the structuring element coincides with a 'black' in the image, make black all pixels from the image covered by the structuring element.

**Notation used for dilation is as follows:**

$$A \oplus B$$

The structuring element can have any shape. Typical shapes are shown in figure 2.5.

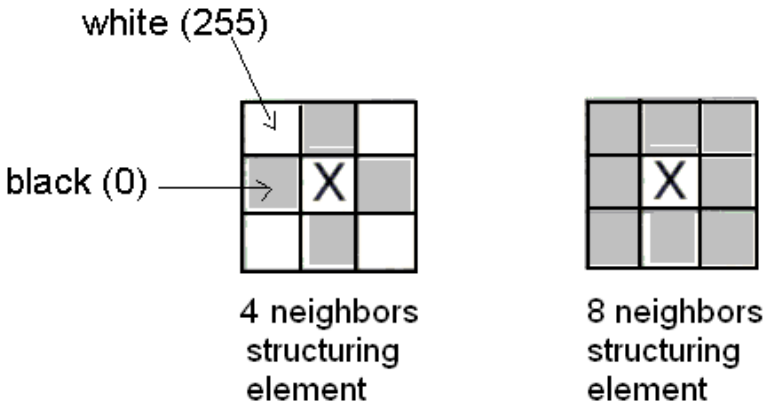


Figure 2. 5 Typical shapes of the structuring elements [15].

### 2.3.4.2 **The erosion**

The erosion process is similar to dilation, but we turn pixels to 'white', not 'black'. As before, slide the structuring element across the image and then follow these steps as follows [15]:

1. If the origin of the structuring element coincides with a 'white' pixel in the image, there is no change; move to the next pixel.

2. If the origin of the structuring element coincides with a 'black' pixel in the image, and at least one of the 'black' pixels in the structuring element falls over a white pixel in the image, then change the 'black' pixel in the image (corresponding to the position on which the center of the structuring element falls) from ‘black’ to a 'white'.

**Notation used for erosion is as follows:**

$$A\Theta B$$

### 2.3.5 Fuzzy erosion and dilation operation

The fuzzy erosion and dilation operations over α-cuts will be defined in this section. First, we have to fuzzify the image pixels to create a fuzzy set image. Using this fuzzy set image, we will define an α-cut from that fuzzy set image as a threshold value. So the image *f(x,y)* will be defined as gα(x). By using this definition the scale of a grayscale image would be selected over the threshold α [14].

If we define a structuring element mask, for example, a 3x3 fuzzified weighted mask μα(x) over the α-cut, then the definition of a 2-D α-cut Bloch and Maitre dilation and erosion with the help of the Minkowski addition and subtraction can be defined as follows[14]:

$$[g(x) \oplus \mu(x)]\ \alpha(x)=supmin[g(x-y),\mu(x)]$$

Where y∈X for Dilation.

$$[g(x) \ominus \mu(x)]\ \alpha(x)=inf\ max[1-g(x-y),\mu(x)]$$

Where y∈X for Erosion.

Figure 2.6 shows the representation of fuzzy morphology dilation formula [14].

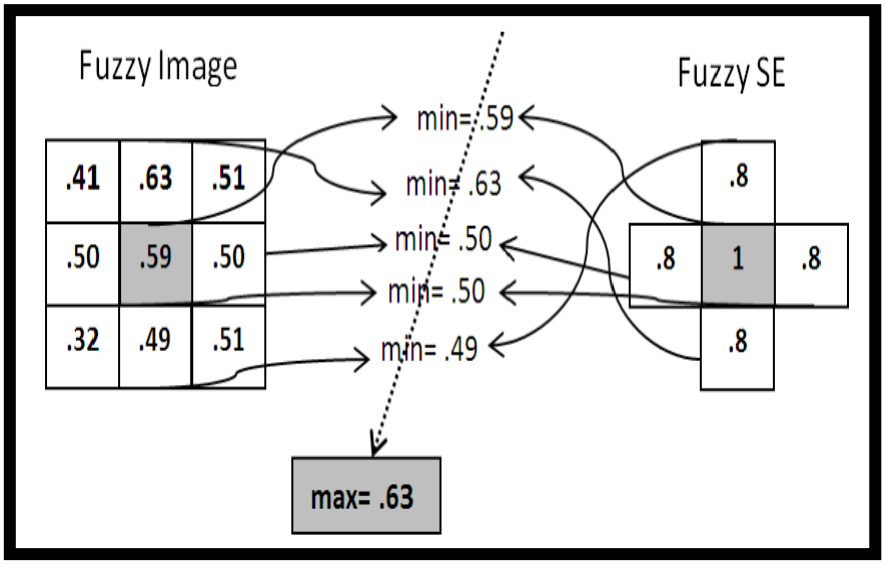


Figure 2. 6. Figural representation of fuzzy morphology formula (Dilation) [14]

## 2.4. Performance Metrics

Image quality assessment (IQA) aims to use computational models to measure the image quality consistently with subjective evaluations. The feature-similarity (FSIM) index of a full reference IQA is based on the fact that human visual system (HVS) understands an image mainly according to its low-level features. Specifically, the phase congruency (PC) and the Gradient magnitude (GM) which are features that represent complementary aspects of the image visual quality. The phase congruency (PC) is a dimensionless measure of the significance of a local structure used as the primary feature in FSIM. Considering that PC is contrast invariant while the contrast information does affect HVS' perception of image quality, and the image gradient magnitude (GM) is employed as the secondary feature in FSIM [16].

### 2.4.1 Mathematics

The computation of FSIM index consists of two stages. In the first stage, the local similarity map is computed, and then in the second stage, we pool the similarity map into a single similarity score, as in the following equation [16]:

$$FSIM = \frac{\sum_{x \in \Omega} S_L(x).PC_m(X)}{\sum_{x \in \Omega} PC_m(X)} \quad 2.1$$

WhereΩmeans the whole image spatial domain, SL is the similarity between two images, and PC is Phase congruency.

### 2.4.2 Justification for the use of FSIM

Image quality assessment (IQA) is very difficult task, widely popular IQA techniques, belonging to objective fidelity like, mean square error (MSE), peak signal to noise ratio (PSNR), or subjective fidelity which corresponding to the human visual system (HVS) like universal quality index (UQI), structural similarity (SSIM), features similarity (FSIM).
T Samajdar, MI Quraishi [66] found that the FSIM is the best metric for the JPEG2000, blur and LAR whereas UQI failed to give better results.

PSNR is useful if images with different dynamic ranges are compared otherwise, it doesn't contain any new information other than MSE. Higher value of PSNR means a good value. PSNR is an excellent measure of quality for white noise distortion.

PSNR involves simple calculations, has clear physical meaning and is convenient in context of optimization but is not according to the characteristics of human visual system (HVS).

UQI is not exactly an HVS model and it performs much better than MSE. It doesn't correlate with subjective assessments hence it is unstable. Wang et al [66] proposed SSIM and FSIM metric involving costly computations. SSIM is a human visual system (HVS) feature based metric. The HVS performs many image processing tasks which are superior to other models. SSIM measures the similarity between two images. It is an improvement over method like MSE and PSNR. It is calculated over several windows of image.

FSIM understand an image based on its low level features like, the phase congruency (PC) and gradient magnitude (GM). PC contains high information and thus is primary feature in FSIM. GM is used to compute gradient magnitude. FSIM is meant for grayscale images or the luminance components of the colored images.

## 2.5. Summary

In this chapter, a look about the technology used to develop the proposed method was presented. The background material required to develop the work undertaken in this dissertation was discussed such as biometric system, fuzzy logic, fuzzy image processing, mathematical morphology, and technologies such fingerprint identification system. Also the fingerprint image quality metric used to measure the quality of fingerprint image was presented.

# CHAPTER III

# Literature Review

In this chapter, introduce the most common existing studies in fingerprint image quality analysis and enhancement.

Fingerprint identification is the most widely used biometric technology and is used in criminal investigations, commercial applications, etc. With such a wide variety of uses for the technology, the demographics and environment conditions that it is used in are just as diverse. However, the identification performance of such system is very sensitive to the quality of the captured fingerprint image. Fingerprint image quality analysis or assessment is useful in improving the performance of fingerprint identification systems. In many systems it is preferable to substitute low quality images for better ones. Therefore, image quality analysis takes an important part in image processing. The quantification of image quality allows to tune a system and to evaluate measurement accuracy of an input image [2].

The principal objective of image enhancement is to process a low quality image input so that the result is a high quality image. The main objective of fingerprint image enhancement is to improve the ridge characteristic features required for minutiae extraction. Ideally, in a well-defined fingerprint image, the ridges and valleys should alternate and flow in a locally constant direction. This regularities facilitates the detection of ridges and consequently allow minutiae to be precisely extracted from the thinned ridges. Thus, the corruption or noise has to be reduced through image enhancement techniques to get enhanced definition of ridges against valleys in the fingerprint image [17].Image enhancement techniques can be divided into three broad categories as follows:

## 3.1. Spatial Domain Method

Refers to the image plane itself. Image processing methods in this category are based on direct manipulation of pixels in an image. Spatial domain can be denoted by the following equation [17]:

$$g(x, y) = T[f(x, y)] \quad 3.1$$

Where *f(x,y)* is the input image, *g(x,y)* is the output image and T is an operator defined over a neighborhood of the point *(x,y).*

Numerous methods have been presented for this domain. Eun-Kyung Yun et al. [6] proposed an adaptive preprocessing method. It is performed through image quality characteristics analysis by extracting features from a fingerprint image and enhancing the images according to their characteristics. Experimental result indicates that the proposed method improved the performance of the fingerprint identification significantly.

Sarrafzadeh, Abdolhossein, FatemehRezazadeh, and JamshidShanbehzadeh [18] introduced an algorithm based on a fuzzy version of brightness preserving dynamic histogram equalization. The algorithm consists of four steps. The first step is the image smoothing, which does employ a Gaussian function for smoothing. This function removes the redundant and noisy maximum and minimum peaks from the image histogram. The second step finds the local maximum points of the histogram by tracing the histogram of the smoothed version of the image. A point on the histogram is a local maximum if its amplitude is more than its neighbors. Next the image histogram is partitioned according to the found maximum points. Each interval is the distance between two successive local maxima. The third step fuzzifies a newly-generated factor from the multiplication of two factors, the interval and the frequency of pixels, in the interval. The third step, at first, finds the length of each interval and the frequency of intensities in that interval. The multiplication of these factors is fuzzified using a triangular shaped membership function, and then employs the TSK model. Step four is Histogram Equalization.

Magudeeswaran, V., and C. G. Ravichandran[19] proposed a fuzzy logic-based on histogram equalization (FHE) for image contrast enhancement to improve the local contrast of the image. The FHE consists of the following operational stages: The first stage, image fuzzification and intensification, where the gray level intensities are transformed to fuzzy plane whose value ranges between 0 and 1. The second stage is the fuzzy histogram computation, where the image is enhanced by making dark pixel darker and making bright pixel brighter and then fuzzy histogram is computed. The third stage is partitioning and equalization of the histogram, where the fuzzy histogram is divided into two sub-histograms based on the median value of the original image and then equalizes them independently to preserve image rightness. The fourth stage is image defuzzification which

is the inverse of fuzzification. The algorithm maps the fuzzy plane back to gray level intensities. Finally, the enhanced image can be obtained.

Selia, I. Golda, And Latha Parthiban[20] proposed two filtering techniques for fingerprint enhancement. In the first one, they used wiener filter to reduce noise which is based on local statistics estimated from a local neighborhood. The second one, they used Gabor filtering technique which did adaptively improve the clarity of ridge and valley structures of input fingerprint images based on the estimated local ridge orientation and frequency.

Hong, Lin, Yifei Wan, and Anil Jain [21] proposed a fast fingerprint enhancement algorithm, which can adaptively improve the clarity of ridge and valley structures of input fingerprint images based on the estimated local ridge orientation and frequency. The main steps of the algorithm included the normalization, local orientation estimation, local frequency estimation, region mask estimation, and filtering.

Balaji, S., And N. Venkatram [22] proposed an integration model of adaptive median filter with directional median filter for fingerprint image enhancement. The adaptive median filter did effectively reduce the impulse noises along the direction of ridge flow. Directional Median Filter (DMF) achieved the follow: Joined the gap with some length between the two ends of broken ridges, cleared the smudges of small size and medium size in the valleys, removed holes in the ridges and smoothed the ridges boundaries.

Sreedhar, K., And B. Panlal [23] proposed methodology that did enhance image characterized by poor contrast and detected the image background. Image enhancement has been carried out by two methods based on the Weber's law notion. The first method employed information from image background analysis by blocks, while the second transformation method utilized the opening/closing operation, which was employed to define the multi-background gray scale images.

Sepasian, M., W. Balachandran, and C. Mares [24] proposed a new two-step methodology to enhance the contrast of the small tiles. In the first step, the contrast limited adaptive histogram equalization (CLAHE) with clip limit was applied to enhance the contrast of the small tiles existing in the fingerprint image and to combine the neighboring tiles using a bilinear interpolation in order to eliminate the artificially induced boundaries.

In the second step, the image was decomposed into an array of distinct blocks and the discrimination of the blocks was obtained by computing the standard deviation of the matrix elements to remove the image background and obtain the boundaries for the region of interest.

Longkumer, Nungsanginla, Mukesh Kumar, A. K. Jaiswal, and Rohini Saxena [25] proposed a novel algorithm using statistical operations and neighborhood processing to image contrast enhancement. The image was subdivided into sectors, which were equalized independently and then using the standard deviation and mean of the pixels in a neighborhood of every point in an image. These two quantities were used to determine the local thresholds, which was used to determine the difference between images. If the difference was greater than a threshold then the original image would be replaced by the equalized image, else the original image would be left as is.

Raju Sonavane, Dr, and B. S. Sawant [26] proposed a special domain fingerprint enhancement methods which did decompose the input fingerprint image into a set of filtered images. From the filtered images, the orientation field was then estimated and a quality mask, which distinguishes the recoverable and unrecoverable corrupted regions in the input image was generated. Using the estimated orientation field, the input fingerprint image was adaptively enhanced in the recoverable regions.

Khalefa, Mustafa Salah, Zaid Amin Abduljabar, and Huda Ameer Zeki [27] proposed an image enhancement method by developing Mehter method for directional image. The enhancement was done by adding the block filtering, histogram equalization and High-Pass filtering. The use of histogram equalization and High-Pass filter did increase the fingerprint contrasts and did enhance ridges and furrows structures to get accurate directional element for each ridges. The use of median filter in block filtering did enable to filter and correct the wrong block direction as well remove noise getting from block wise operation.

Lu, Haiping, Xudong Jiang, and Wei-Yun Yau [28] proposed a novel fingerprint image post-processing algorithm. It was developed based on several rules, which were generalized through a study on the errors that commonly occur in minutiae extraction and their effects on the overall verification performance. The algorithm did trace adaptively fingerprint image, and each valid ridge was associated with a ridge number. If a tracing

ridge number intersects another traced ridge, then a bifurcation would be detected. An ending would be detected if the tracing of a ridge number would stop with no other ridge intersection.

Kumar, U. Pavan, and P. Padmaja [29] developed an image enhancement algorithm that modified the local contrast and the local luminance mean of an image and controlled the local contrast as a function of the local luminance mean of the image. The algorithm first separates an image into its lows (low pass filtered form) and highs (high pass filtered form) components. The lows component then controls the amplitude of the highs component to increase the local contrast. The lows component was then subjected to a non-linearity to modify the local luminance mean of the image and was combined with the processed highs component.

Chaurasia, Om Preeti [30] proposed an algorithm which was able to estimate all the intrinsic properties of the fingerprints such as the foreground region mask, local ridge orientation and local ridge frequency. The steps used in the pre-processing were Binarization, thinning, dilation, thinning of the dilated image, removing unwanted portions from the image (Refining), and producing dual image.

Naresh Kumar, and Parag Verma [31] used CLAHE to enhance the contrast of small tiles and to combine the neighboring tiles in an image by using bilinear interpolation which eliminates the artificially induced boundaries.

Hanoon, Muna F [32] proposed a low bit rate image coding scheme based on vector quantization. The block prediction coding techniques were employed to cut down the required bit rate of vector quantization. In block prediction coding, neighboring encoded blocks were taken to compress the current block. The four steps procedure for fingerprint compression and enhancement used the following: CLAHE, Wiener filtering, image Binarization, thinning and Vector Quantization (VQ).

Kanpariya Nilam, and Rahul Joshi [33] proposed a methodology of automatic parameter selection for fingerprint enhancement procedures. The enhanced performances were not satisfactory because of the complicated ridge and valley structures that were affected by unusual input contexts that needed adaptive tuning of parameters. The methods for constructing enhancement image for fingerprint images was normalization, orientation estimation, ridge frequency estimation, and filtering.

Sang Keun Oh, Joon Jae Lee, Chul Hyun Park, Bum Soo Kim, and Kil Houm Park [34] proposed a method of fingerprint image enhancement using the Directional Filter Bank (DFB). It was achieved by handling the directional sub band images effectively using the directionality of the fingerprint image. The enhanced image was reconstructed by synthesizing the subband images followed by subband processing, in which a few essential subband components were strengthened while others were weakened.

Table 3.1 shows a brief survey of image enhancement techniques in spatial domain.

Table 3.1. A Brief Survey of Image Enhancement Techniques in spatial domain

| Author | Year | Model | Processing technique description |
|---|---|---|---|
| Hong | 1998 | Fast algorithm | Estimated local ridge orientation and frequency and then applied normalization, Local orientation estimation, Local frequency estimation, Region mask estimation, and Filtering. |
| Lu | 2002 | post processing algorithm | The algorithm traces fingerprint image, and each valid ridge is associated with a ridge number. If a tracing ridge number is intersects another traced ridge, a bifurcation is detected. An ending is detected when the tracing of a ridge number is ended with no other ridge intersection. |
| Sang | 2003 | Directional Filter Bank (DFB) | Handling the directional sub band images effectively using the directionality of the fingerprint image. Reconstructed image by synthesizing the sub band images. |
| Yun, Eun-Kyung | 2006 | adaptive pre-processing | Extracts five features from a fingerprint image, analysis image quality with clustering method, and enhance the image according to its characteristics using adaptive pre-processing method. |
| Raju Sonavane | 2007 | Special Domain | Decompose fingerprint image into a set of filtered images. From the filtered images estimated the orientation field and a quality mask. By estimated orientation, the fingerprint image is adaptively enhanced. |
| Sepasian | 2008 | Contrast Limited Adaptive Histogram Equalization (CLAHE) | Firstly, CLAHE with clip limit is applied to enhance the contrast of the small tiles and combine the neighboring tiles using a bilinear interpolation. In a second step, the image is decomposed into an array of distinct blocks and the discrimination of the blocks is obtained by computing the standard deviation of the matrix elements to remove the image background and obtain the boundaries for the region of interest. |
| Balaji | 2008 | Integration Model Of Adaptive Median Filter With Directional | Adaptive median filter are used, which can effectively reduce impulse noises along the direction of ridge flow by Directional Median Filter (DMF). It can Join the gap with some length between the two ends of broken ridges, clear the smudges of small size and medium size in the valleys, |

| | | | |
|---|---|---|---|
| | | Median Filter | remove holes in the ridges and smooth the ridges boundaries. |
| Khalefa | 2011 | Mehtre Method | Use of Histogram equalization and High-Pass Filter to increase fingerprint contrasts and enhance ridges. The use of Median Filter in Block Filtering enables to filter and correct wrong block direction and removes noise getting from Block Wise Operations. |
| Selia | 2011 | filtering techniques | Two filters, the first one uses Wiener filter. The second one use Gabor filtering technique. |
| Hanoon | 2011 | Contrast Limited Adaptive Histogram Equalization With Clip Limit (CLAHE) | Applied the following steps Wiener filtering, image Binarization, thinning and Vector Quantization (VQ). |
| Sreedhar | 2012 | Weber's law notion and transformation | Employs information from image background analysis by blocks. The transformation method is employed to define the multi-background grey scale images. |
| Chaurasia | 2012 | Morphological Operations | Estimate all the intrinsic properties of the fingerprint then apply the pre-processing step, Binarization, thinning, dilation, thinning of the dilated image, removing unwanted portions from the image (Refining), and producing a dual image. |
| Naresh | 2012 | Contrast Limited Adaptive Histogram Equalization (CLAHE) | Used to enhance the contrast of small tiles and combine the neighboring tiles in an image by using bilinear interpolation. |
| Kumar | 2013 | New Algorithm | Separate an image into its low and high components. The low components controls the amplitude of the high components to increase the local contrast. The low components are then subjected to a non-linearity to modify the local luminance mean of the image and is combined with the processed high components. |
| Longkumer | 2014 | Histogram Equalization | Divided an image into equalized sectors then use the standard deviation and mean to determine local thresholds that can determine the difference between images. |
| Kanpariya | 2014 | Adaptive Tuning Parameters | The method apply the following steps: Normalization, Orientation Estimation, Ridge Frequency Estimation, and Filtering. |

### 3.2. Frequency Domain Method

This method consists of modifying the Fourier transform of an image and then computing the inverse transform, Discrete Fourier Transform (DFT), to get back the input image. Thus given a digital image *f(x,y),* of size M x N. The basic filtering equation has the following form [17]:

$$g(x,y) = \Im - 1\,[\,H(u,v)F(u,v)] \qquad 3.2$$

Where $\Im - 1$ is the Inverse Discrete Fourier Transform (IDFT), F (u,v) is the DFT of the input image *f(x,y), H(u,v)* is the filter function and g(x,y) is the filtered output image. Specification of H(u,v) is simplified considerably by using functions that are symmetric about the center. This was accomplished by multiplying the input image by (-1) x+y prior to computing its transform [17].

Deshmukh, Rucha Ashok, And S. A. Ladhake [35] proposed a method to enhance the contrast of color image by using Laplacian pyramid, which could enhance the brightness and contrast of an input image by decomposing it into band pass images and perform contrast as well as detail enhancement on each of these images. When the individual images were enhanced in contrast and detailed manner, then by combining them together the enhanced output image would be generated.

Bhowmik, Pankaj, Kishore Bhowmik, Mohammad Nurul Azam, and Mohammed Wahiduzzaman Rony [36] combined two image enhancement processes, the Discrete Fourier Transform, which was used to decompose an image into its sine and cosine components, and the Histogram Equalization, which did increase the global contrast of many images, especially when the usable data of the image was represented by close contrast values.

Chikkerur, Sharat, Venu Govindaraju, and Alexander N. Cartwright [37] proposed a new approach for fingerprint enhancement based on Short Time Fourier Transform (STFT) Analysis. The algorithm simultaneously estimates all the intrinsic properties of the fingerprints such as the foreground region mask, local ridge orientation and local frequency orientation. The image was then divided into overlapping windows, and assumed that the image was stationary within this small window and could be modeled approximately as a surface wave. The Fourier spectrum of this small region was analyzed and probabilistic

estimates of the ridge frequency and ridge orientation were obtained and in each window a filter was applied.

JitendraChoudhary, Dr.Sanjeev Sharma and Jitendra Singh Verma [38] proposed a fingerprint image enhancement method which could adaptively improve the clarity of ridge and furrow structures of an input fingerprint image based on the following: frequency and spatial domain filtering, local orientation estimation, local frequency estimation and morphological operation. The main steps of the method were preprocessing, compute the discrete Fourier transform, convolution with High Pass Filter, compute the inverse discrete Fourier transform, apply the quick mask edge detector to the inverse DFT image, apply the average filter using convolution for smooth the ridges, normalization, local orientation estimation, local frequency estimation, region mask estimation, Gabor Filtering, post-processing, and enhanced image.

T. Vidhya and T. K. Thivakaran [39] proposed an approach to enhance fingerprint images, which was done using both Gabor filter method and wavelet de-noising method. The wavelet de-noising which enhances the fingerprint image by reducing noise and selecting threshold values by the computation of discrete wavelet transform. The method deals with the fingerprint images affected by two types of noises namely Gaussian noise and salt &pepper noise.

Patil, Vikas D., and Sachin D. Ruikar [40] proposed a methodology of image enhancement that uses principle component analysis (PCA) in wavelet domain. The method consists of two stages. In the first stage, Local Pixel Grouping algorithm (LPG) was applied as well as the PCA on the blur image. The random noise was significantly reduced in the first stage. In the second stage,the wavelet thresholding methods for removing random noise was applied, which has been researched extensively due to its effectiveness and simplicity.

Narayana, Battula RVS, and K. Nirmala [41] proposed a technique that used Discrete Wavelet Decomposition (DWT) to decompose an image into different subbands, and then the high frequency subband images have been interpolated. The interpolated high frequency subband coefficients have been corrected by using the high frequency subbands achieved by Stationary Wavelet Transform (SWT) of the input image. An original image was interpolated with half of the interpolation factor used for interpolation the high

frequency subbands. Afterwards, all these images have been combined using inverse DWT to generate a super resolved imaged.

Mohammed sayeemuddin, Shaikh, Sima K. Gonsai, and Dharmesh Vandra [42] proposed a method to enhance fingerprint image based on Gabor filter. The steps used in Gabor filter were normalization, local ridge orientation, local ridge frequency, Gabor filtering based on local ridge orientation and frequency. An overlapping block-wise filtering was used instead of conventional pixel wise Gabor filtering. An average frequency of entire image was used instead of each pixel frequency.

Babatunde, Iwasokun G., Akinyokun O. Charles, Alese B. Kayode, and Olabode Olatubosun [43] proposed a modified version of an existing mathematical algorithm for improving the quality of fingerprint image. The modified algorithm consists of sub-models for fingerprint segmentation, normalization, ridge orientation estimation, ridge frequency estimation, Gabor filtering, Binarization and thinning.

V.Aarthy, R. Mythili, and M.Mahendran[44] proposed a Fast Fourier Transform algorithm (FFT) with two-stage scheme to enhance the low-quality of fingerprint image in both the spatial domain and the frequency domain based on the learning from the images. The first-stage enhancement scheme has been designed to use the context information of the local ridges to connect or separate the ridges, based on spatial filtering, therefore, the broken ridges will be connected and the merged ridges will be separated effectively. In the second-stage processing, the filter was separable in the radial and angular domains, respectively.

Table 3.2 shows a brief survey of image enhancement techniques in frequency domain.

Table 3.2. Brief Survey of Image Enhancement Techniques in Frequency domain

| Author | Year | Model | Processing technique description |
|---|---|---|---|
| Chikkerur | 2005 | Short Time Fourier Transform (STFT) Analysis | Estimates all the intrinsic properties of the fingerprints then the image is divided into overlapping windows. The Fourier spectrum of this small region is analyzed and probabilistic estimates of the ridge frequency and ridge orientation are obtained and in each window apply a filter. |
| Jitendra | 2011 | Frequency And Spatial Domain | Pre-processing, Compute the discrete Fourier transform, , Compute the Inverse discrete Fourier transform, inverse DFT image, average filter, |

| | | | |
|---|---|---|---|
| | | Filtering | Normalization, Local orientation Estimation, Local Frequency Estimation, Region Mask Estimation, Gabor Filtering, and Post-processing. |
| Bhowmik | 2012 | Fourier Transform and Histogram Equalization | FT used to decompose an image into its sine and cosine, and the HE, increases the global contrast of images. |
| T. Vidhya | 2012 | Gabor Filter And Wavelet De-noising | The wavelet de-noising which enhances the fingerprint image by reducing noise and selecting threshold values by computation of discrete wavelet transform. |
| Patil | 2012 | Principle Component Analysis (PCA) and Wavelet | Apply Local Pixel Grouping algorithm (LPG) and PCA on the blur image and apply the wavelet thresholding methods. |
| Narayana | 2012 | Discrete Wavelet Decomposition (DWT) | Decompose an image into different sub bands. The high frequency sub band interpolated and its coefficients are corrected by using the high frequency sub bands achieved by Stationary Wavelet Transform (SWT). |
| Babatunde | 2012 | Gabor filtering | The algorithm consists of sub-models for fingerprint segmentation, normalization, ridge orientation estimation, ridge frequency estimation, Gabor filtering, Binarization and thinning. |
| V.Aarthy | 2012 | Fast Fourier Transform algorithm (FFT) | Two stages, the first-stage is an enhancement scheme which use the context information of the local ridges to connect or separate the ridges, based on spatial filtering. The second-stage is processing step where the filter is separable to the radial and angular domains. |
| Deshmukh | 2013 | Laplacian pyramid | Decomposing images into band pass images and perform contrast on each of these images. When the individual images are enhanced in contrast and detailed manner, then combine them together to get the enhanced output image. |
| Mohammed | 2014 | Gabor filter | The steps used are the following: normalization, local ridge orientation, local ridge frequency, and Gabor filtering based on local ridge orientation and frequency. |

### 3.3. Fuzzy Domain Method

Fuzzy domain method is based on gray level mapping into a fuzzy plane, using a membership transformation function. The aim is to generate an image of higher contrast than the original image by giving larger weights to the gray levels that are closer to the mean gray level of the image than to those that are farther from mean. An image I of size M x N and L gray level can be considered as an array of fuzzy singletons, each having a value of membership denoting its degree of brightness relative to some brightness level [45].

In last decade, there were many proposed fuzzy image enhancement methods. TaruMahashwari and Amit Asthana [12] proposed a fuzzy if then rules method, which was applied on grayscale image by converting the image data into fuzzy domain using Gaussian function and modify membership function by using if then rules as flow:

“If pixel intensity is dark then output is darker”, or “if pixel intensity is gray then output is gray”, or “if pixel intensity is bright then output is brighter”. The last stage is the defuzzification, which the inverse of fuzzification the algorithm that maps the fuzzy plane back to gray level intensities.

Hasikin, Khairunnisa, and Nor Ashidi Mat Isa [46] proposed contrast enhancement technique, which minimized fuzziness in the image without requiring a complex procedure and long computational time. They fuzzified the original image by using the S-function and specified its three parameters (a,b and c) to ensure the membership function maximizes the information contained in the image. It was done by incorporating two fuzzy measures, namely fuzzy entropy and index of fuzziness. The membership function transformed the image intensity levels from the spatial domain to fuzzy domain and classified the image to overexposed and underexposed by using exposure parameter which denote percentage of the image gray levels and then threshold was determined to divide the image into two parts.

Cheng, Heng-Da, and Huijuan Xu [47] proposed a novel adaptive direct fuzzy contrast enhancement method based on the fuzzy entropy principle and fuzzy set theory, which did combine the local contrast measurement with contour detection operator. Maximum fuzzy entropy principle was used to map an image from space domain to fuzzy domain by using the S-membership function. The method was able to automatically determine the related parameters of the S-membership function according the nature of the

image. The last stage was the defuzzification, which did transform the modified membership value to gray level.

Kundra, Harish, ErAashima, and Er Monika Verma [48] proposed an algorithm which did remove the noise and improve the contrast of the image by using fuzzy logic control approach. In the first stage, the impulse noise was removed by a constructed filter, which replaced the central pixel in the window of the image, which did maximize the sum of the similarities between all its neighbors. The second stage was improving the contrast of the image by setting the shape of membership function regarding to the actual image by setting the value of fuzzifier, therefore, modified the membership values by using linguistic hedges, and at the end generated the new gray levels image.

Selvi, M., and Aloysius George [49] proposed a method which was designed to identify the noise pixels area and enhance that portion a lone by using fuzzy based filtering technique and adaptive thresholding. The method applied in four stages. The first stage was the preprocessing, which cropped the original database images into specified size. The second stage was the fuzzy filtering technique, which made a filter by replacing the central pixel in the window of the image by that one which maximizes the sum of similarities between all its neighbors. The third stage was adaptive thresholding, which determined the threshold value and this value was applicable to use with the image having different types of noises which selected for each type of noise. The last stage was the morphological operation, which did perform two operation, the dilation process and the erosion process, which used to eliminate all pixels in regions that are too small to contain the structuring image.

DiwakarShrivastava, VineetRichhariya [50] proposed a novel adaptive fuzzy contrast enhancement technique based on the fuzzy entropy principle and fuzzy set theory. It maps an image from spatial domain to fuzzy domain by using the S-membership function, which it automatically determine the related parameters according to the nature of the image. The modified membership values and the defuzzification are then applied to transform the membership values to the gray level.

Suple, Nutan Y., and Sudhir M. Kharad [51] proposed a method to enhance image contrast using fuzzy inference system. The method is made of four stages; morphological processing, the process of convert the image data from crisp values to fuzzy values by

using membership functions, and make the defuzzification process, which converts the image fuzzified values to crisp values.

SJ, MrsPreethi, and Mrs K. Rajeswari [45] proposed a ramp membership function and modification of membership function using square and cube operators for enhancing the visual appearance of digital image using fuzzy logic based approach. The square and cube operators did modify the membership function by making the dark pixel darker by decreasing its fuzzy bright membership degree, and the one in the middle was not altered much and the pixels that was bright was made brighter by increasing its fuzzy bright membership degree.

M.Suneel, K.Samba Siva Rao, M.Lavanya and M.SaiSasanka, [52] proposed an image enhancement method through the use of fuzzy logic based S-shaped membership function for contrast enhancement of X-ray image in medical diagnosis. The method suppresses noise or any other fluctuations in the image in the smoothing operation by using median filtering which reduce the blurring of the image edges and calculate the fuzzy property plane by using S-shaped membership function and then transform the fuzzy property plane using intensification operator. The improved image can be obtained by using gray transformation.

MadasuHanmandlu and DevendraJha [53] proposed the use of Gaussian membership function to fuzzify the image information in spatial domain. They introduce a global contrast intensification operator (GINT), which contains three parameters; intensification parameter (t), fuzzifier (fh), and the crossover point (μc), for enhancement of color images. They defined fuzzy contrast-based quality factor (Qf) and entropy-based quality factor (Qe) and the corresponding visual factors for the desired appearance of images. By minimizing the fuzzy entropy of the image information with respect to these quality factors, the parameters t, fh and μc were calculated globally. By using the proposed technique, a visible improvement in the image quality was observed for under exposed images, as the entropy of the output image was decreased. The terminating criterion was decided by both the visual and quality factors. For over exposed and under exposed images, the proposed fuzzification function needs to be modified by taking the maximum intensity as the fourth parameter. The type of the images was indicated by the visual factor which was less than 1 for under exposed images and more than 1 for over exposed images.

Popa, Camelia, MihaelaGordan, AurelVlaicu, BogdanOrza, and Gabriel Oltean [54] proposed an implementation strategy suitable for single input – single output Takagi-Sugeno fuzzy systems with trapezoidal shaped input membership function, for JPEG image in compressed domain. The fuzzy set parameters were adaptively chosen by analyzing the histogram of the image data in the compressed domain, in order to optimally enhance the image contrast. The fuzzy rule-based algorithm required some threshold comparisons, for which an adaptive implementation, taking into account the frequency content of each block in the compress domain JPEG image was proposed. This guarantees the minimal error implementation at minimum computational cost.

Perumal, S. Arumuga, T. C. Rajakumar, and N. Krishnan [55] proposed a fuzzy modeling approach which was used for reducing the noise and increasing the brightness of the ridges. The Fuzzy statistical values were determined on the basis of the gray intensity of the spatial domain. The image intensity was transformed with the specified Probability Density Function (PDF). The Cumulative Density Function (CDF) was used in accumulating pixel values in an ordered manner for stretching the contrast. The Fuzzy statistical value of the input image and the inverse transform were used for histogram mapping.

Choi, YoungSik, and Raghu Krishnapuram [56] proposal incorporated a robust adaptive filtering into a fuzzy rule-based system that did smooth noise while preserving edges and image details. They derived three different filters for these purposes using the weighted (or fuzzy) least squares (LS) method. Each filter was applied when certain conditions were satisfied. The first filtering system was most effective for removing impulse noise while preserving fine details, and outperformed the others when the noise level was small. The second filtering system outperformed the others with regard to the root mean square (RMS) criterion when images were severely contaminated and could be used for smoothing noise while preserving edges. Third filtering system was somewhat less effective for noise attenuation. To evaluate the situation at a particular pixel, we used the total compatibility of the neighboring pixels with the center pixel. The total compatibility was the mean value of the membership degrees to which the neighboring pixels represented a center pixel in a given window. They constructed three different fuzzy rule-based filtering

systems and made “soft” decisions based on each condition, aggregated the decisions made, and finally made a decision based on the aggregation.

Vlachos, Ioannis K., and George D. Sergiadis [57] presented an automated algorithm for contrast enhancement, based on a class of parametric indices of fuzziness. This class of indices has the ability of adjusting the location of the maximum in any point of the [0, 1] interval and thus modeling the ambiguity and vagueness of images in a more adaptive way. In order to derive classes of parametric indices of fuzziness, parameter-dependent complements were used and extending to maximizing the index of fuzziness of an image. The modification of the membership values of image pixels in the fuzzy plane was performed according to an S-function, which maximized the appropriate parametric index of fuzziness (PIF). The selection of the optimal parameter of the PIF was carried out by exploiting the statistics of the image in the fuzzy domain, using its first-order fuzzy moment.

Khandelwal, Manglesh, Shweta Saxena, and Priya Bharti [58] proposed a new approach to enhance color image corrupted by Gaussian noise known as Fuzzy Filter (FZF). The proposed approach did start as follow: input to the system the original image, convert it into RGB plane, add Gaussian noise to the original image, produce the fuzzy logic rules, RGB values of image will be passed to fuzzy logic, fuzzy output will differentiate between ambiguous colors, and compare the results with existing methods.

Table 3.3 shows a brief survey of image enhancement techniques in fuzzy domain.

Table 3.3. A Brief Survey of Image Enhancement Techniques in fuzzy domain

| Author | Year | Model | Processing technique description |
|---|---|---|---|
| Choi, YoungSik | 1997 | adaptive filtering with fuzzy rule-based system | Smooth noise, and derive three different filters. Each filter is applied when certain conditions are satisfied. They constructed three different fuzzy rule-based filtering systems. |
| Cheng | 2000 | Novel Adaptive Direct Fuzzy Contrast Enhancement | Used maximum fuzzy entropy principle to map an image to fuzzy domain by using the S- membership. Modification of the membership value was performed, then transform the modified membership value to grey level with defuzzification. |
| Khandelwal | 2005 | Fuzzy Filter | Input original image to the system, convert it into RGB plane, adding Gaussian noise to the original image, |

| | | | |
|---|---|---|---|
| | | | produce fuzzy logic rules. RGB values of image will be passed to fuzzy logic, fuzzy output will differentiate among ambiguous colors, then compare the results with existing methods |
| MadasuHanmandlu | 2006 | Gaussian Membership Function | They introduced GINT, which contains three parameters; intensification parameter (t), fuzzifier (fh), and the crossover point (μc), for enhancement of color images. They defined fuzzy contrast-based quality factor (QF) and entropy-based quality factor (QE) as visual factors for the desired appearance of images. By minimizing the fuzzy entropy of the image information with respect to these quality factors the parameters t, fh and μc are calculated globally. |
| Vlachos, Ioannis K. | 2006 | automated algorithm | Class of parametric indices of fuzziness - modify the membership values with S-function, select optimal parameter of PIF by exploiting the statistics of the image in the fuzzy domain. |
| Popa, Camelia | 2008 | Takagi-Sugeno with trapezoidal membership function | The fuzzy sets parameters are adaptively chosen by analyzing the histogram of the image data in the compressed domain. |
| Kundra | 2009 | Fuzzy Logic Control Approach | Constructed filter, setting the shape of membership function, modification of membership values by using linguistic hedge, defuzzification. |
| Perumal, S | 2011 | Fuzzy Modeling Approach | The Fuzzy statistical values are determined for the image and then intensity is transformed with the specified Probability Density Function (PDF). The Cumulative Density Function (CDF) is used in accumulating pixel values in an ordered manner for stretching the contrast. |
| Hasikin | 2012 | Fuzzy Greyscale Enhancement Technique | Fuzzify image with S-membership function modification, exposure parameter denote percentage of the image grey levels and then threshold is determined to divide the image into two parts overexposed and underexposed |
| TarunMahashwari | 2013 | Fuzzy Inference system | Convert the image data into fuzzy domain using Gaussian function and modify membership function using if then rules and get new membership values by using square function, then defuzzify. |
| Suple | 2013 | Fuzzy Inference System | The method consists of morphological processing, fuzzify image values with membership function, and modify the membership values and defuzzification. |
| SJ, Mrs | 2013 | Ramp | Use the square and cube operators to modify the |

| | | | |
|---|---|---|---|
| | | Membership Function | membership function by making the dark pixel darker, and the one in the middle is not altered much and the pixels that is bright is made brighter. |
| M.Suneel | 2013 | Fuzzy Logic Based S-Shaped Membership Function | Enhancement of X-ray image by smoothing operation with median filtering, S-shaped membership function, and then transform the fuzzy property plane using intensification operator. |
| Diwakar | 2014 | Novel Adaptive Fuzzy Contrast Enhancement Technique | Use S-membership function and fuzzy entropy principle to map an image to fuzzy domain, modification the membership values, then defuzzify. |

### 3.4. Image Quality Analysis

Fingerprint image quality analysis or assessment is useful to improve the performance of fingerprint recognition systems. In many systems it is preferable to substitute low quality images for better ones. Therefore, image quality analysis may take an important part in image processing. The quantification of image quality allows to tune a system and to evaluate measurement accuracy on the given image.

Alexey Saenko et al, [59] proposed a method to analysis image quality which uses two ways. Crisp method uses a matrix norm which is a scalar that gives some measure of the magnitude of the elements of the matrix. There are different types of matrix norm such as.

- The I-norm or largest column sum of M, max (sum (abs (M))).
- The largest singular value.
- The infinity norm, or largest row sum of M, max (sum (abs (M’))).
- The frobenius norm of matrix M, sqrt(sum(diag(M’*M)));

On the other hand, fuzzy logic method evaluates the image quality by using If-Then rules with the following parameters:

- Sharpness: amount of pixels on the threshold of objects.
- Noise: arithmetical mean of gray level.
- Contrast: difference between maximal and minimal gray levels.

- Vignetting: difference between gray levels at the edge and center of images.
- Field curvature: difference between sharpness values at the edge and center of images.

All of these parameters are linguistic variables and have three possible values “bad”, “normal”, and “good”. The image quality is “bad’ if these parameters have big values and the image quality is “good” if these parameters have small values or equal to 0.

Eun-Kyung Yun et al. [6] proposed a method to analysis image quality by extracted five features from fingerprint. Therefore, according to these features, images are clustered by using ward’s clustering algorithm which is one the hierarchical clustering methods initially assigns an independent cluster to each sample. Then it seeks the most similar pairs of clusters and merges them into one cluster.

Syam, Rahmat, MochamadHariadi, and Mauridhi Hery Purnomo [60] proposed a model to determine the standard value used to classifying the type of distortion in the fingerprint images based on the image quality. They used the ridge-valley clarity score and ridge-valley thickness ratio score to describe the fingerprint image quality. The model computed the standard value of acquisition distortion parameters of fingerprint images as follow:

The first stage is the segmentation, which is the process of separating the foreground regions in the image from the background regions. The second stage is the normalization which is the process to reduce the variation of gray level values along the ridges and valleys. The operation is pixel-wise and does not change the clarity of the ridges and valleys. The third stage is analysis of ridge-valley clarity which is the method of analyzing the distribution of segmented ridge and valley. The fourth is the ridge-valley thickness ratio, which is the process to compute the ratio for ridge thickness to valley thickness. And finally the stage of analysis of parameter values, which analysis ridge-valley by setting the parameter value in oily, dry, and neutral fingerprint images based on the clarity score and the ridge-valley thickness ratio of ridge-valley. Once the local clarity score, global clarity score, and ridge-valley thickness ratio are obtained, the average of each parameter is calculated, then the maximum value and the minimum value of parameters are computed.

Syam, Rahmat, Mochamad Hariadi, and Mauridhi Hery Purnomo[61] proposed a novel procedures to determine the parameter values of dry fingerprint images based on the score of clarity and ridge-valley thickness ratio. The parameters are local clarity scores, global clarity scores and ridge-valley thickness ratio. The analysis started by quantizing fingerprint images into blocks with size of32x32 pixels. The orientation of each block was perpendicularly calculated to the ridge. The middle of the block along the ridge with two-dimensional vector (V1)of the size 32x13 pixels was extracted and transformed into a two-dimensional vertical vector (V2). Linear regression was applied to the one-dimensional vector (V3), which is the average of vector (V2) to produce a Determinant Threshold (DT1). Less than area of DT1 is called a ridge, while the opposite is a valley. The tests carried out by calculating the clarity of the image from the overlapping area of the gray-level distribution of ridge and valley that has been separated. The thickness ratio of ridge to valley was then computed for each block based on gray-level value per block of image in the normal direction toward the ridge. Finally, the thickness ratio of ridge to valley for all images are found from which the average value is obtained.

Kukula, Eric, Stephen Elliott, Hakil Kim, and Cristina San Martin [62] proposed a study to quantitatively analyze the impact of fingerprint pressing force on both image quality and the number of detected minutiae. Two experiments were performed with a cross Match VerifierTM 300 LC single optical fingerprint capture device to measure the impact fingerprint force has on image quality. Investigating the quality scores of experiment further, one can deduce that the quality scores significantly increase between the 3N and 5N-7N force levels. The quality regressed at 11N, but there was minimal benefit of applying more than 9N of force, as the quality scores did not improve by much, plus were deemed as neutral or unsatisfactory by the users.

Li, Qing-Rong, And Mei Xie [63] proposed a method for measuring fingerprint image quality using Fourier spectrum. This method did measure fingerprint image quality based on the global characteristics of the image and did not relay on local ridge orientation estimation. First the band frequency which corresponds to the global average period of ridge was searched. Then the quality score of the fingerprint image was computed by measuring relative magnitude of the band frequency components.

Modi, Shimon K., and Stephen J. Elliott [64] studied the impact of fingerprint image quality of two different age groups: 18-25, and 62 and above on overall performance using two different matchers. The difference in image quality between the two age groups was analyzed, and then the impact of image quality on performance of fingerprint matchers between the two groups was analyzed. Image quality analysis was performed using NFIQ which is part of NIST Fingerprint Image Software (NFIS). Neurotechnologija VeriFinger and bozorth3 (NFIS) matchers were used to assess overall performance.

### 3.5. Open Issues

From the literature review and previous studies, it found that the fingerprint image enhancement is still a hot area of research because it has many problems and open issues which needs to be addressed such as:

- The diseases of hand and finger (Hand eczema, Fingertip eczema, Pyoderma, Systemic sclerosis) cause problems for the most types of sensors, because the color of the skin and structure of epidermis and dermis are influenced.
- User skin conditions and placement pressure such as dry, oily, and damp skin can degrade the quality of the live scan, as can surface contaminants or variability in pressure applied to the sensor by the user which effect in performance of the matching minutiae.
- For older people the collagen level in the skin is reduced enough to cause complications in fingerprint scan reliability which can affect the performance of the matching minutiae.
- For children the fingerprint is changed from childhood to adult age, which causes problems for people registry system in many countries.

### 3.6. Summary

This chapter reviewed all the main methods in the fingerprint image quality analysis and enhancement research area. In particular, three categories are outlined for image enhancement: (i) Spatial domain method (ii) Frequency domain method (iii) Fuzzy domain method. Also reviews of the open issues in this research area was also presented.

# CHAPTER IV

## The Methodology

In this chapter, the methodology used to perform fingerprint image quality analysis and enhancement will be presented.

The proposed research is grouped into the following subsections. First, we will present fingerprint image quality analysis using fuzzy inference system methods. Secondly, we will employ fuzzy morphology on dry and oily fingerprint images to enhance them, which will be done after image quality analysis. Figure 4.1 shows the flowchart for the whole method used to analyze fingerprint image quality as well as enhancement.

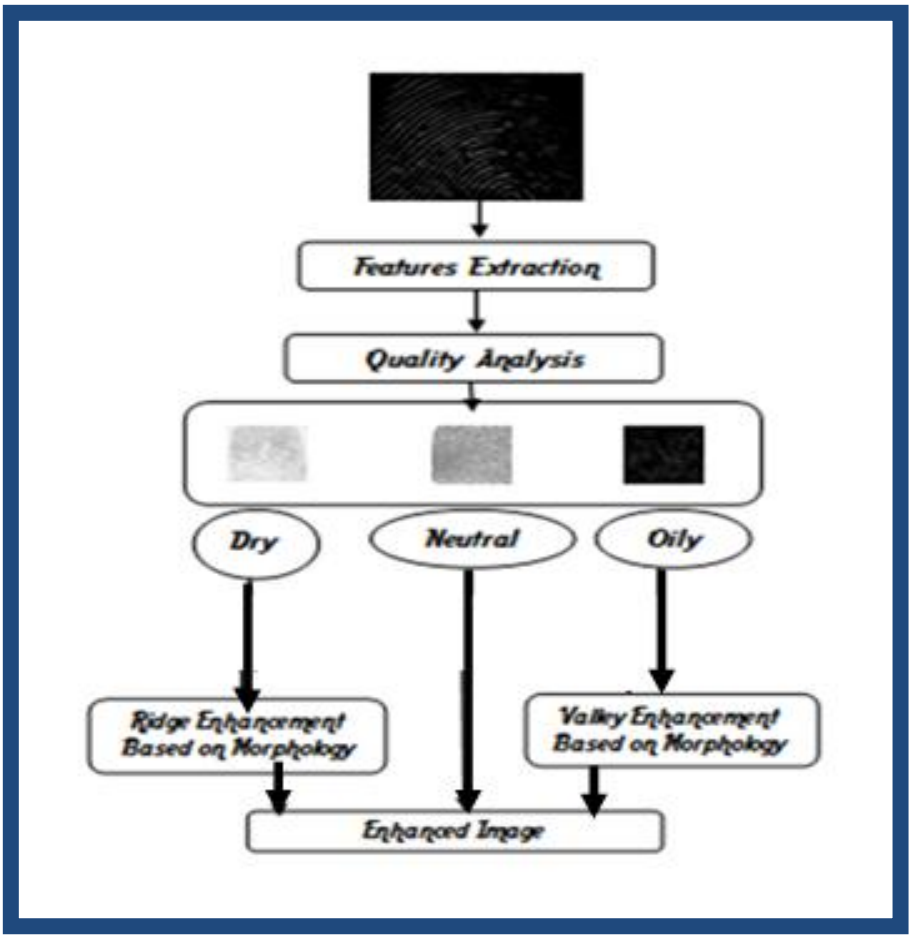


Figure 4.1.The flowchart for the whole method.

To show the proposed research methodology in detail, first, the fingerprint image quality analysis is described to determine the distortion of each image. Secondly, the morphology operations is applied to enhance the ridge for dry images and valley for oily images.

Finally, performance evaluation is conducted to compare the Features Similarity index (FSIM) for the proposed and existing methods.

## 4.1. Fingerprint image quality analysis

The first stage in the proposed research method is the fingerprint image quality analysis. In this stage, the quality of fingerprint image is analyzed into three types dry, neutral and oily, according to their features. This is performed in order to determine the image type and as well the suitable method to enhance it.

Figure 4.2 shows the flowchart of the proposed method. The method starts by extracting four features from the fingerprint image for image quality assessment using a fuzzy inference system to determine the type of the fingerprint image quality.

### 4.1.1. Features Extraction

In this stage we extracted four features which are the Local Clarity Score (LCS), Global Clarity Score (GCS), Ridge Valley Thickness Ratio (RVTR), and the Global Contrast Factor (GCF), to analyze the fingerprint image quality using a fuzzy inference system. The following subsections will present few fingerprint image scores that will be used in analyzing fingerprint images.

#### 4.1.1.1. Ridge-Valley Clarity Scores of Fingerprint Images

Ridge and valley clarity analysis indicates the ability to distinguish the ridge and valley along the ridge direction. A method of analyzing the distribution of segmented ridge and valley is introduced to describe the clarity of the given fingerprint pattern [60].

To perform local clarity analysis, the fingerprint image is quantized into blocks of size 32x32 pixels. Inside each block, an orientation line, which is perpendicular to the ridge direction, is computed. At the center of the block along the ridge direction, a 2D vector $V_1$ (slanted square shown in Figure 4.3) with size 32x13 pixels can be extracted and transformed to a vertical aligned 2-D Vector $V_2$. Equation (1) is a 1-D Vector $V_3$, which is the average profile of $V_2$ [60].

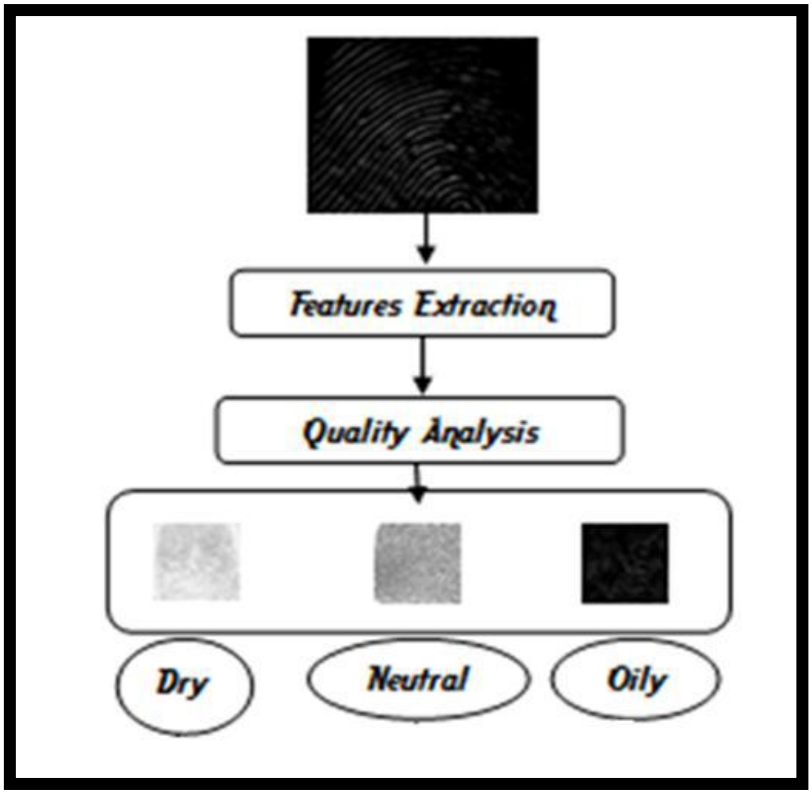


Figure 4.2.The flowchart for the new proposed algorithm for fingerprint image quality analysis

$$V_3(i,j) = \frac{\sum_{j=1}^{m} v_2}{m}, i = 1 \dots 32 \quad (4.1)$$

Where *m* is the block height (13 pixels) and *i* is the horizontal index. Once $V_3$ is calculated as in (4.1), then linear regression can be applied to $V_3$ to find the Determine Threshold (DT1). Figure 4.3 shows the method of regional segmentation. DT1 is the line positioned at the center of the vector $V_3$ and is used to classify the ridge region and valley region. Regions lower than DT1 are classified as the ridges and the others are as valleys. Hence, the regions of ridges and valleys can be separated in the 2-D vector $V_2$ by the 1-D average profile $V_3$ with the DT1 shown as the dotted straight line in Figure 4.4. As ridges and valleys are separated, a clarity test can be performed in each segmented 2-D rectangular regions. Figure 4.5 shows the gray level distribution of the segmented ridges and valleys. The overlapping area is the region of misclassification, which is the area of failing to determine ridge or valley accurately by using DT1. Hence, the area of the overlapping region can be an indicator of the clarity of ridge and valley. For the calculation of the clarity score refer to equations (2), (3), and (4) [60].

$$\alpha = \frac{V_B}{V_T} \tag{4.2}$$

$$\beta = \frac{\Re_B}{\Re_T} \tag{4.3}$$

$$LCS = \frac{(\alpha + \beta)}{2} \tag{4.4}$$

Where $V_B$ is the number of bad pixels in the valley that the intensity is lower than the $DT_1$. $V_T$ is the total number of pixels in the valley region. $\Re_B$ is the number of bad pixels in the ridge that the intensity is higher than the $DT_1$ and $\Re_T$ is the total number of pixels in the ridge region. $\alpha$ and $\beta$ are the portion of bad pixels. Hence, the Local Clarity Score (LCS) is the average value of both α and $\beta$.

For ridges with good clarity, both distributions should have a very small overlapping area. The following factors affect the size of the total overlapping area [60]:

1. Noise on ridge and valley.
2. Scar across the ridge pattern.
3. Water patches on the image due to wet finger.
4. Incorrect of orientation angle due to the effect of directional noise.
5. Highly curved ridge.
6. Minutiae, bifurcation, delta point or core.

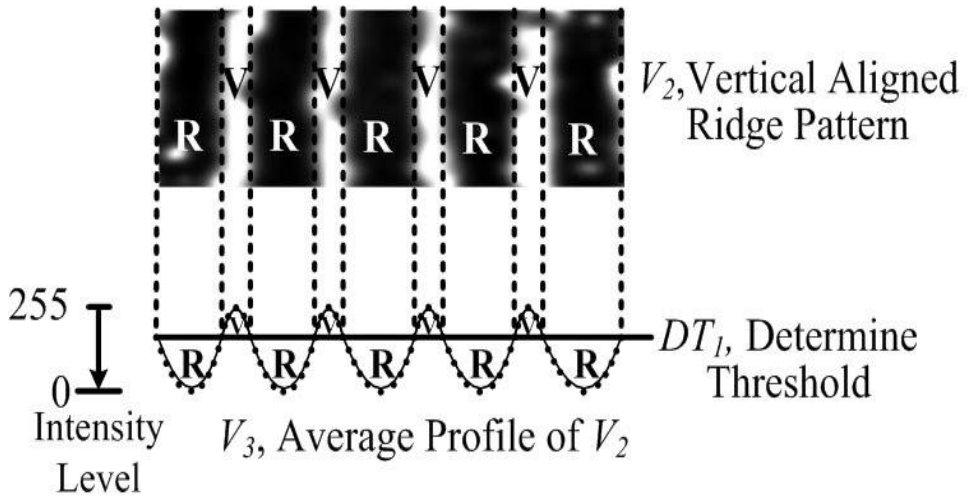


Figure 4.3.Region Segmentation of vector V2 [60].

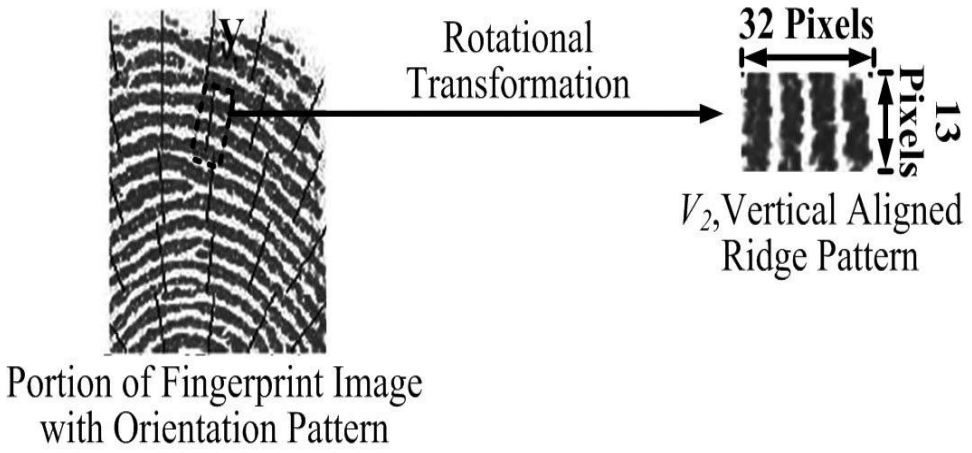


Figure 4.4. Extraction of a local region and transformation to vertical aligned ridge pattern [60].

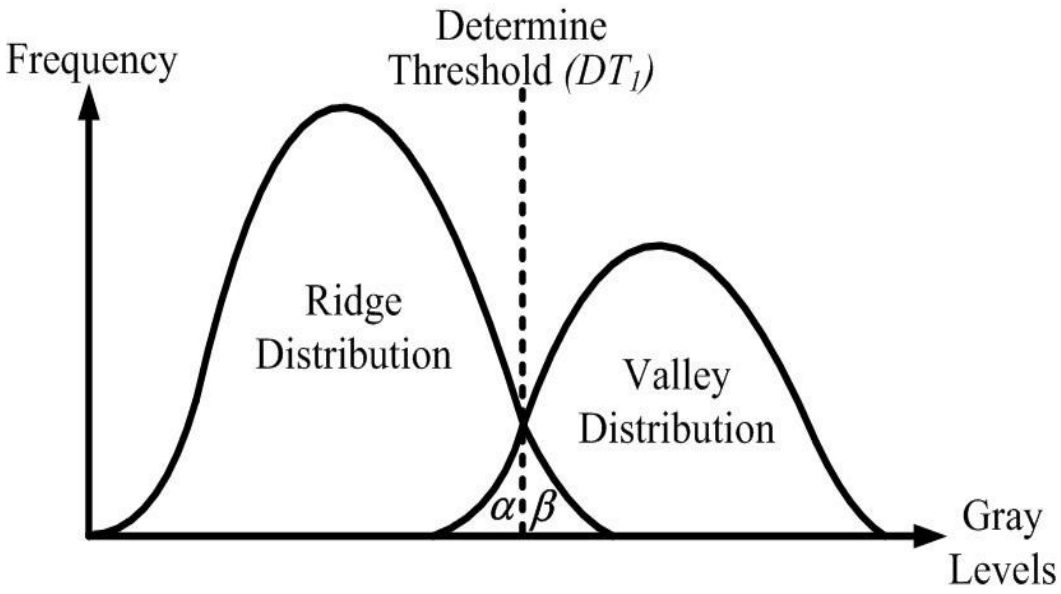


Figure 4.5.Distribution of ridge and valley [60].

Factors 1 to 4 are physical noise found in the image. Factors 5 and 6 are actual physical characteristics of the fingerprint. Therefore, a small window with size 32x13 pixels is chosen to minimize the chance of encountering too many distinct features in the same location. The Global Clarity Score (*GCS*) can be computed by the expected value of the *LCS* [60].

$$GCS = E(LCS(i,j)) \tag{4.5}$$

Where

$$E(.) = \sum_{i=1}^{H}\sum_{j=1}^{v}(.) \tag{4.6}$$

As seen in equation (5), LCS (i, j) is the Clarity Scores which is calculated according to equations (2), (3) and (4) at location (i, j), where i and j are horizontal and vertical index of the image block, respectively. H and V are the maximum number of horizontal and vertical blocks, respectively. The GCS can be used to describe the general ridge clarity of a given fingerprint image.

**4.1.1.2.The ratio for ridge thickness to valley thickness is computed in each block.**

The thickness of ridge and valley are obtained using gray level values for one image block in the direction normal to the flow ridge. Later, the ratio of each block is computed and average value of the ratio is obtained over the whole image [60].

**4.1.1.3.Global Contrast Factor (GCF)**

Contrast in image processing is usually defined as a ratio between the darkest and the brightest spots of an image. The newly introduced Global Contrast Factor (GCF) corresponds closer to the human perception of contrast. GCF uses contrasts at various resolution levels in order to compute overall contrast [65].

The contrast of any (small) part of an image is called the local contrast. The global contrast is defined as the average local contrast of smaller image fractions [65].

First compute the local contrast factors at various resolutions, and then build a weighted average based on human perception method which it can be approximated with the square root of the linear luminance, which it gamma corrected luminance using a gamma of 2.2 for standard displays [65].

Let us denote the original pixel value with k, $k \in \{0, 1\ldots 254, 255\}$. The first step is to apply gamma correction with $Y$=2.2, and scale the input values to the [0, 1] range. We will denote the scaled and corrected values linear luminance with L [65].

$$L = \left(\frac{K}{255}\right)^{Y} \tag{4.7}$$

The perceptual luminance L is now

$$L = 100 * \sqrt{L} \quad = 100\sqrt{\left(\frac{K}{255}\right)^{Y}} \tag{4.8}$$

The square root to compute luminance was used [65]. Once the perceptual luminance are computed we have to compute local contrast. For each pixel we compute the average difference of L between the pixel and four neighboring pixels [65].

By assuming the image is w pixels wide and h pixels high, and the image is organized as a one dimensional array of row-wise sorted pixels and the local contrast $Lc_i$ for pixel i is [65]:

$$Lc_i = \frac{|L_i - L_{i-1}| + |L_i - L_{i+1}| + |L_i - L_{i-w}| + |L_i - L_{i+w}|}{4} \quad 4.9$$

The average local contrast for current resolution $C_i$ is computed now as the average local contrast $Lc_i$over the whole image [65].

$$C_i = \frac{1}{w*h} * \sum_{i=1}^{w*h} LC_i \quad 4.10$$

Now that we have computed average local contrasts $C_i$, we need to find the weigh factors $w_i$, which will be used to compute the global contrast factor as follows:

$$GCF = \sum_{i=1}^{N} W_i * C_i \quad 4.11$$

### 4.1.2. Fuzzy Inference System

Fuzzy inference system (FIS) will be used to analysis fingerprint image quality. FIS has five stages. The stages are as follow:

#### 4.1.2.1. The Fuzzy Inference System Editor

This stage shows the fuzzy inference system editor which displays the general information about the system.

#### 4.1.2.2. The Membership Function Editor

This stage shows the membership function for all features; Local Clarity Score (LCS), Global Clarity Score (GCS), Ridge Valley Thickness Ratio (RVTR) and the Global Contrast Factor (GCF),

#### 4.1.2.3. The Rules Editor

This stage shows the constructing rule base for four features, which will used as inputs and having three outputs (Dry, Neutral and Oily).

#### 4.1.2.4. The rule viewer

The rule viewer**,** which display a roadmap of the whole fuzzy inference process, shows the values of the input variables and their outputs, and has ability to change inputs and see output changes**.**

#### 4.1.2.5. The surface viewer

The surface viewer presents three dimensional curve that represent input features and the image quality type.

## 4.2. Fingerprint Image Enhancement

Fingerprint image enhancement (FIE) is an essential preprocessing step in fingerprint recognition applications. The Image enhancement is a preprocessing technique to make the image clearer than the original image. FIE is needed to increase the contrast between ridges and furrows and for connecting the false broken points of ridges due to insufficient amount of ink and other reasons. The noises and degradations arise due to mainly reasons such as poor skin conditions, varying finger pressure while acquisition, sensor noise and dry, wet or dirty fingers. They may be degraded and corrupted with elements of noise due to many factors including variations in skin and impression conditions. High quality fingerprint image is very important for fingerprint verification to work properly. Therefore, the main objective of fingerprint enhancement techniques to process an input image and make it more suitable for identification.

### 4.2.1 Enhancing Dry fingerprint image

Dry fingerprint image is the image that has ridges that are scratchy locally and there are many white pixels in the ridges, which arise due to dry skin that tends to cause inconsistent contact of the finger ridges with the scanner's platen surface, causing broken ridges and many white pixels replacing ridge structure.

To enhancement ridge of dry fingerprint image, the following stages are applied as shown in figure 4.6:

A. Smoothing: is applied to the original image to reduce noise in the image and to decrease the disparity between pixel values by using low pass filter.
B. Dilation: image is dilated to increase the valleys and enlarges the width of maximum regions. Dilation is applied using fuzzy on smoothing fingerprint image in stages as follows:
    1. Fuzzify the input image (smoothing fingerprint image). For image fuzzification we are required to employ membership function where the membership values lie between 0 and 1.

2. Apply the α-cut dilation (Fuzzy dilation).
3. Defuzzify which is the inverse of fuzzification process.

C. The final stage, the union operation stage, where a logical operation exist that extracts the union of black pixels in an original image (input dry image) and the dilated image. In this stage the white pixels in the ridges are removed and we obtain an enhanced image with good ridges.

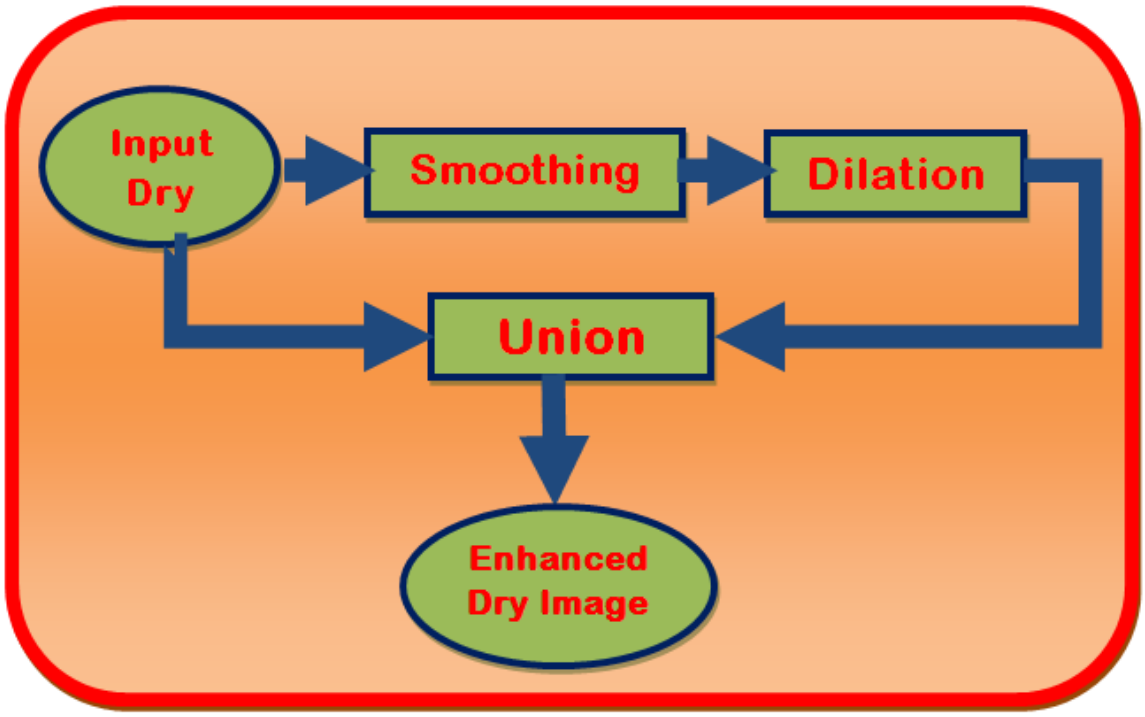


Figure 4.6.The flowchart for the proposed enhancement ridge of dry fingerprint image method.

### 4.2.2 Enhancing Oily Fingerprint Image

Oily fingerprint image is an image that has partial separation of ridges and valleys, but some parts of valleys are filled up causing them to appear dark or adjacent ridges stand close to each other in many regions. Therefore, their ridges tend to be very thick. They arise due to oily skin which tend to fill up with moisture, causing them to appear black in the image similar to ridge structure.

To enhance the valley of oily fingerprint image, the following stages are applied as shown in figure 4.7:

A. Smoothing: is applied to original image to eliminate thin and separated valleys by using low pass filter.
B. Fuzzy Dilation: dilated image is obtained and it contains the regions when ridges are sufficiently separated as blocks and regions where ridges touch one another as white. Dilation is applied using fuzzy morphology on smoothing fingerprint image in stages as follows:
    1. Fuzzify the input image (smoothing fingerprint image). For image fuzzification we are required to employ membership function where the membership values lie between 0 and 1.
    2. Apply the α-cut dilation (Fuzzy dilation).
    3. Defuzzify which is inverse of fuzzification process.
C. Composition of black pixels in the original image and in the image obtained in step B.
D. Fuzzy erosion is applied on the original image using fuzzy morphology in stages as follows:
    1. Fuzzify the original input image. For image fuzzification we are required to employ membership function where the membership values lie between 0 and 1.
    2. Apply the α-cut erosion (Fuzzy erosion).
    3. Defuzzify which is inverse of fuzzification process.
E. Composition of block pixels using fuzzy erosion of an image (step D) and the inverse of an image in step B.
F. Extracting the union of block pixels of the image is obtained in steps C and E.
G. By applying all of these stages, the distorted fingerprint imagewill be enhanced.

### 4.3. The Database

We will verified the proposed method using the DB_ITS_2009 database[60] which is a private database collected by the Department of Electrical Engineering, Institute of Technology Sepuluh Nopember Surabaya. The database was taken with great caution because of the image quality considerations. The DB_ITS_2009 database was taken using an optical sensor U.are.U 4000B fingerprint reader with the specifications: 512 dpi,

USB 2.0, flat fingerprint and uncompressed. This database has 1704 fingerprint images of size 154x208 pixels. The details are as follows:

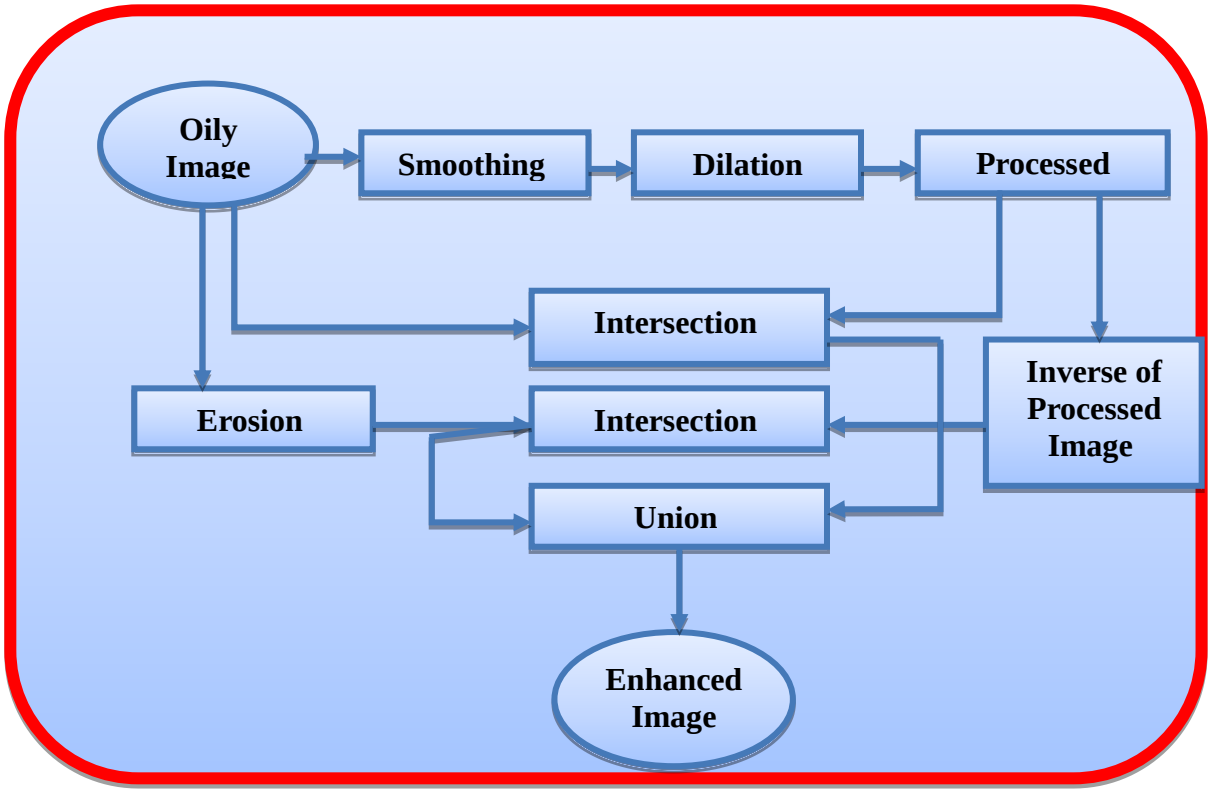


Figure 4.7.The flowchart for the proposed oily enhancement method.

The fingerprint images are classified into three types the finger conditions (dry, neutral and oily). Each type of finger condition consists of 568 fingerprint images sourced from 71 different fingers. Each of these fingerprint images was taken eight times for the three conditions above. As a result, we obtained 3x71x8=1074 fingerprint images. To obtain dry fingerprint images, hair-dryer was used to completely dry the fingertip. Likewise, in order to get oily fingerprint images, we smeared baby-oil on the fingertips before the image was taken [60].

### 4.4. Performance Measurement

To measure the fingerprint image quality with subjective evaluations for both the proposed and existing methods we use feature-similarity (FSIM) index. Full reference Image Quality Assessment (IQA) is based on the fact that human visual system (HVS) understands an image mainly according to its low-level features. Features such as the phase

congruency (PC) and the gradient magnitude (GM), which represent complementary aspects of the image visual quality. The PC is a dimensionless measure of the significance of a local structure, which is used as the primary feature in FSIM. Considering that PC is contrast invariant while the contrast information does affect HVS' perception of image quality, and the GM is employed as the secondary feature in FSIM [16].

### 4.5. Summary

This chapter described the methodology that was used to analysis fingerprint image quality as well as image enhancement. Fuzzy inference system with the four features; GCS, LCS, RVTR, and GCF was used to analysis fingerprint image. Fuzzy morphology was also used to enhance the dry and oily fingerprint images.

# CHAPTER V
# Fingerprint Image Quality Analysis

In general, the fingerprint image quality relies on the clearness of separated ridges by valleys and the uniformity of the separation. Although the change in environmental conditions such as temperature, humidity and pressure might influence a fingerprint image in many ways, but the condition of skin still dominate the overall quality of the fingerprint image. The proposed method starts by extracting four features from the fingerprint image for quality assessment using a Fuzzy Inference System to determine the type of the fingerprint image quality. In this chapter, the methodology used to perform fingerprint image quality analysis will be described.

## 5.1. Ridge-Valley Clarity Scores of Fingerprint Images

Ridge and valley clarity analysis indicates the ability to distinguish the ridge and valley along the ridge direction. A method of analyzing the distribution of segmented ridge and valley is introduced to describe the clarity of the given fingerprint image.

For the method to compute the Global Clarity Score (*GCS*) and the Local Clarity Score (LCS), refer to equations (2), (3), and (4) in chapter IV [60], there the description of the whole method used to compute the LCS and GCS, by applied this method with DB_ITS_2009 database [60] (see appendix A) we obtained the following data. Table 5.1 shows the GCS for dry fingerprint image samples. Table 5.2 shows the GCS neutral fingerprint image samples. Table 5.3 shows the GCS oily fingerprint image samples. Table 5.4 shows the LCS for dry fingerprint image samples. Table 5.5 shows the LCS for neutral fingerprint image samples. Table 5.6 shows the LCS for oily fingerprint image samples.

Table 5.1.The GCS for Dry Fingerprint Images.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0118 | 0.0117 | 0.0119 | 0.0122 | 0.0120 | 0.0118 | 0.0118 | 0.0119 |
| 2. | 0.0118 | 0.0119 | 0.0118 | 0.0119 | 0.0119 | 0.0119 | 0.0119 | 0.0119 |
| 3. | 0.0119 | 0.0117 | 0.0118 | 0.0118 | 0.0116 | 0.0117 | 0.0119 | 0.0117 |
| 4. | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0118 | 0.0118 | 0.0120 |
| 5. | 0.0118 | 0.0117 | 0.0117 | 0.0119 | 0.0117 | 0.0117 | 0.0118 | 0.0117 |
| 6. | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 |
| 7. | 0.0118 | 0.0117 | 0.0118 | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0118 |
| 8. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 9. | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0118 | 0.0117 |
| 10. | 0.0118 | 0.0118 | 0.0120 | 0.0119 | 0.0118 | 0.0120 | 0.0119 | 0.0118 |
| 11. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 12. | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0117 |
| 13. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0118 |
| 14. | 0.0119 | 0.0117 | 0.0118 | 0.0117 | 0.0116 | 0.0118 | 0.0118 | 0.0117 |
| 15. | 0.0119 | 0.0120 | 0.0120 | 0.0119 | 0.0120 | 0.0120 | 0.0118 | 0.0119 |
| 16. | 0.0116 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0118 | 0.0121 | 0.0119 |
| 17. | 0.0119 | 0.0121 | 0.0120 | 0.0119 | 0.0120 | 0.0121 | 0.0121 | 0.0119 |
| 18. | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0115 | 0.0117 | 0.0118 |
| 19. | 0.0123 | 0.0122 | 0.0121 | 0.0123 | 0.0124 | 0.0125 | 0.0122 | 0.0121 |
| 20. | 0.0121 | 0.0120 | 0.0119 | 0.0121 | 0.0121 | 0.0122 | 0.0121 | 0.0122 |
| 21. | 0.0117 | 0.0119 | 0.0118 | 0.0118 | 0.0116 | 0.0118 | 0.0117 | 0.0117 |
| 22. | 0.0121 | 0.0122 | 0.0123 | 0.0123 | 0.0121 | 0.0121 | 0.0122 | 0.0121 |
| 23. | 0.0116 | 0.0119 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0116 | 0.0119 |
| 24. | 0.0118 | 0.0118 | 0.0119 | 0.0119 | 0.0119 | 0.0121 | 0.0121 | 0.0120 |
| 25. | 0.0117 | 0.0118 | 0.0118 | 0.0120 | 0.0117 | 0.0117 | 0.0118 | 0.0117 |
| 26. | 0.0116 | 0.0118 | 0.0118 | 0.0118 | 0.0118 | 0.0117 | 0.0117 | 0.0117 |
| 27. | 0.0119 | 0.0117 | 0.0118 | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0117 |

Table 5.2. GCS neutral fingerprint image

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0119 | 0.0118 | 0.0118 | 0.0120 | 0.0118 | 0.0119 | 0.0119 | 0.0119 |
| 2. | 0.0119 | 0.0117 | 0.0118 | 0.0119 | 0.0118 | 0.0119 | 0.0118 | 0.0117 |
| 3. | 0.0119 | 0.0116 | 0.0118 | 0.0120 | 0.0116 | 0.0115 | 0.0118 | 0.0117 |
| 4. | 0.0117 | 0.0119 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0117 |
| 5. | 0.0118 | 0.0117 | 0.0117 | 0.0120 | 0.0117 | 0.0116 | 0.0116 | 0.0118 |
| 6. | 0.0118 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0118 | 0.0116 | 0.0116 |
| 7. | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0118 | 0.0119 | 0.0119 | 0.0118 |
| 8. | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0116 |
| 9. | 0.0115 | 0.0115 | 0.0114 | 0.0115 | 0.0115 | 0.0114 | 0.0116 | 0.0115 |
| 10. | 0.0119 | 0.0118 | 0.0119 | 0.0120 | 0.0119 | 0.0119 | 0.0118 | 0.0119 |
| 11. | 0.0118 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0117 |
| 12. | 0.0118 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0116 | 0.0116 |
| 13. | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0117 | 0.0116 | 0.0117 |
| 14. | 0.0120 | 0.0117 | 0.0116 | 0.0116 | 0.0115 | 0.0115 | 0.0115 | 0.0115 |
| 15. | 0.0121 | 0.0118 | 0.0120 | 0.0120 | 0.0119 | 0.0119 | 0.0117 | 0.0118 |
| 16. | 0.0119 | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 |
| 17. | 0.0120 | 0.0120 | 0.0120 | 0.0119 | 0.0119 | 0.0118 | 0.0119 | 0.0118 |
| 18. | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0118 | 0.0116 | 0.0117 | 0.0117 |
| 19. | 0.0118 | 0.0121 | 0.0120 | 0.0121 | 0.0121 | 0.0123 | 0.0121 | 0.0123 |
| 20. | 0.0122 | 0.0121 | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0121 |
| 21. | 0.0118 | 0.0118 | 0.0118 | 0.0120 | 0.0121 | 0.0120 | 0.0118 | 0.0117 |
| 22. | 0.0123 | 0.0120 | 0.0120 | 0.0120 | 0.0119 | 0.0120 | 0.0119 | 0.0119 |
| 23. | 0.0116 | 0.0116 | 0.0115 | 0.0115 | 0.0116 | 0.0116 | 0.0116 | 0.0118 |
| 24. | 0.0120 | 0.0119 | 0.0120 | 0.0121 | 0.0120 | 0.0122 | 0.0123 | 0.0119 |
| 25. | 0.0119 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 |
| 26. | 0.0118 | 0.0117 | 0.0117 | 0.0119 | 0.0119 | 0.0120 | 0.0117 | 0.0118 |
| 27. | 0.0116 | 0.0115 | 0.0116 | 0.0115 | 0.0116 | 0.0115 | 0.0115 | 0.0115 |

Table 5.3. GCS for oily fingerprint image.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0121 | 0.0126 | 0.0121 | 0.0121 | 0.0123 | 0.0125 | 0.0122 | 0.0123 |
| 2. | 0.0123 | 0.0123 | 0.0124 | 0.0122 | 0.0122 | 0.0121 | 0.0124 | 0.0122 |
| 3. | 0.0120 | 0.0122 | 0.0120 | 0.0121 | 0.0120 | 0.0123 | 0.0123 | 0.0121 |
| 4. | 0.0122 | 0.0122 | 0.0123 | 0.0121 | 0.0122 | 0.0122 | 0.0122 | 0.0121 |
| 5. | 0.0120 | 0.0121 | 0.0122 | 0.0121 | 0.0122 | 0.0120 | 0.0121 | 0.0120 |
| 6. | 0.0120 | 0.0122 | 0.0122 | 0.0122 | 0.0121 | 0.0120 | 0.0122 | 0.0123 |
| 7. | 0.0124 | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0121 |
| 8. | 0.0122 | 0.0122 | 0.0121 | 0.0122 | 0.0123 | 0.0122 | 0.0124 | 0.0124 |
| 9. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0121 |
| 10. | 0.0123 | 0.0122 | 0.0122 | 0.0121 | 0.0123 | 0.0123 | 0.0121 | 0.0121 |
| 11. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0120 | 0.0121 | 0.0123 | 0.0120 |
| 12. | 0.0120 | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0120 |
| 13. | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 |
| 14. | 0.0122 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0120 | 0.0120 | 0.0121 |
| 15. | 0.0127 | 0.0128 | 0.0126 | 0.0126 | 0.0127 | 0.0126 | 0.0123 | 0.0121 |
| 16. | 0.0121 | 0.0122 | 0.0122 | 0.0121 | 0.0121 | 0.0122 | 0.0122 | 0.0121 |
| 17. | 0.0124 | 0.0123 | 0.0121 | 0.0121 | 0.0124 | 0.0122 | 0.0123 | 0.0123 |
| 18. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0120 | 0.0120 | 0.0121 | 0.0120 |
| 19. | 0.0127 | 0.0125 | 0.0123 | 0.0127 | 0.0126 | 0.0128 | 0.0128 | 0.0127 |
| 20. | 0.0127 | 0.0126 | 0.0128 | 0.0126 | 0.0126 | 0.0127 | 0.0126 | 0.0129 |
| 21. | 0.0128 | 0.0127 | 0.0129 | 0.0125 | 0.0125 | 0.0125 | 0.0126 | 0.0124 |
| 22. | 0.0128 | 0.0130 | 0.0129 | 0.0125 | 0.0127 | 0.0125 | 0.0125 | 0.0125 |
| 23. | 0.0123 | 0.0122 | 0.0121 | 0.0121 | 0.0126 | 0.0122 | 0.0124 | 0.0122 |
| 24. | 0.0122 | 0.0125 | 0.0126 | 0.0124 | 0.0122 | 0.0122 | 0.0125 | 0.0123 |
| 25. | 0.0127 | 0.0124 | 0.0125 | 0.0125 | 0.0125 | 0.0125 | 0.0125 | 0.0125 |
| 26. | 0.0122 | 0.0121 | 0.0122 | 0.0120 | 0.0121 | 0.0122 | 0.0122 | 0.0121 |
| 27. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0119 | 0.0120 | 0.0120 | 0.0121 |

Table 5.4. LCS for dry fingerprint image.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0129 | 0.0133 | 0.0126 | 0.0124 | 0.0158 | 0.0127 | 0.0119 | 0.0152 |
| 2. | 0.0127 | 0.0129 | 0.0126 | 0.0144 | 0.0127 | 0.0126 | 0.0138 | 0.0127 |
| 3. | 0.0130 | 0.0127 | 0.0120 | 0.0124 | 0.0129 | 0.0126 | 0.0138 | 0.0126 |
| 4. | 0.0132 | 0.0132 | 0.0132 | 0.0133 | 0.0132 | 0.0134 | 0.0130 | 0.0126 |
| 5. | 0.0128 | 0.0132 | 0.0127 | 0.0132 | 0.0129 | 0.0128 | 0.0127 | 0.0130 |
| 6. | 0.0136 | 0.0133 | 0.0133 | 0.0133 | 0.0134 | 0.0137 | 0.0131 | 0.0134 |
| 7. | 0.0127 | 0.0129 | 0.0122 | 0.0123 | 0.0129 | 0.0130 | 0.0128 | 0.0137 |
| 8. | 0.0132 | 0.0134 | 0.0136 | 0.0133 | 0.0133 | 0.0133 | 0.0135 | 0.0134 |
| 9. | 0.0125 | 0.0129 | 0.0130 | 0.0131 | 0.0128 | 0.0135 | 0.0126 | 0.0129 |
| 10. | 0.0128 | 0.0130 | 0.0125 | 0.0127 | 0.0130 | 0.0127 | 0.0128 | 0.0130 |
| 11. | 0.0135 | 0.0136 | 0.0134 | 0.0132 | 0.0134 | 0.0133 | 0.0132 | 0.0131 |
| 12. | 0.0131 | 0.0131 | 0.0130 | 0.0128 | 0.0130 | 0.0130 | 0.0131 | 0.0129 |
| 13. | 0.0128 | 0.0127 | 0.0129 | 0.0131 | 0.0128 | 0.0130 | 0.0128 | 0.0139 |
| 14. | 0.0130 | 0.0129 | 0.0129 | 0.0131 | 0.0132 | 0.0130 | 0.0129 | 0.0130 |
| 15. | 0.0168 | 0.0141 | 0.0141 | 0.0153 | 0.0137 | 0.0153 | 0.0140 | 0.0160 |
| 16. | 0.0131 | 0.0130 | 0.0131 | 0.0129 | 0.0131 | 0.0130 | 0.0171 | 0.0129 |
| 17. | 0.0130 | 0.0128 | 0.0139 | 0.0139 | 0.178 | 0.0151 | 0.0130 | 0.0137 |
| 18. | 0.0130 | 0.0123 | 0.0129 | 0.0127 | 0.0129 | 0.0134 | 0.0131 | 0.0129 |
| 19. | 0.0131 | 0.0163 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0131 |
| 20. | 0.0132 | 0.0129 | 0.0131 | 0.0131 | 0.0144 | 0.0132 | 0.0132 | 0.0132 |
| 21. | 0.0132 | 0.0176 | 0.0126 | 0.0124 | 0.0131 | 0.0129 | 0.0127 | 0.0139 |
| 22. | 0.0161 | 0.0165 | 0.0161 | 0.0161 | 0.0172 | 0.0161 | 0.0161 | 0.0184 |
| 23. | 0.0131 | 0.0173 | 0.0132 | 0.0133 | 0.0140 | 0.0132 | 0.0132 | 0.0157 |
| 24. | 0.0150 | 0.0130 | 0.0150 | 0.0133 | 0.0133 | 0.0218 | 0.0209 | 0.0213 |
| 25. | 0.0131 | 0.0131 | 0.0130 | 0.0124 | 0.0134 | 0.0131 | 0.0130 | 0.0131 |
| 26. | 0.0135 | 0.0133 | 0.0130 | 0.0132 | 0.0133 | 0.0134 | 0.0134 | 0.0137 |
| 27. | 0.0125 | 0.0132 | 0.0129 | 0.0128 | 0.0131 | 0.0129 | 0.0131 | 0.0131 |

Table 5.5. LCS for neutral fingerprint image.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0124 | 0.0125 | 0.0129 | 0.0165 | 0.0129 | 0.0133 | 0.0123 | 0.0132 |
| 2. | 0.0129 | 0.0131 | 0.0128 | 0.0128 | 0.0131 | 0.0130 | 0.0129 | 0.0132 |
| 3. | 0.0131 | 0.0132 | 0.0131 | 0.0189 | 0.0134 | 0.0134 | 0.0174 | 0.0132 |
| 4. | 0.0126 | 0.0128 | 0.0129 | 0.0127 | 0.0129 | 0.0127 | 0.0130 | 0.0128 |
| 5. | 0.0129 | 0.0129 | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0126 |
| 6. | 0.0132 | 0.0132 | 0.0128 | 0.0134 | 0.0132 | 0.0127 | 0.0133 | 0.0132 |
| 7. | 0.0135 | 0.0117 | 0.0134 | 0.0135 | 0.0122 | 0.0126 | 0.0135 | 0.0126 |
| 8. | 0.0131 | 0.0132 | 0.0131 | 0.0132 | 0.0131 | 0.0131 | 0.0131 | 0.0130 |
| 9. | 0.0134 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0133 | 0.0133 | 0.0132 |
| 10. | 0.0125 | 0.0129 | 0.0130 | 0.0124 | 0.0128 | 0.0131 | 0.0127 | 0.0126 |
| 11. | 0.0133 | 0.0133 | 0.0131 | 0.0130 | 0.0131 | 0.0133 | 0.0130 | 0.0130 |
| 12. | 0.0131 | 0.0128 | 0.0130 | 0.0131 | 0.0128 | 0.0129 | 0.0130 | 0.0131 |
| 13. | 0.0132 | 0.0131 | 0.0130 | 0.0131 | 0.0130 | 0.0129 | 0.0132 | 0.0130 |
| 14. | 0.0163 | 0.0132 | 0.0132 | 0.0133 | 0.0134 | 0.0131 | 0.0131 | 0.0130 |
| 15. | 0.0135 | 0.0134 | 0.0139 | 0.0165 | 0.0140 | 0.0134 | 0.0132 | 0.0170 |
| 16. | 0.0132 | 0.0132 | 0.0129 | 0.0131 | 0.0130 | 0.0131 | 0.0135 | 0.0131 |
| 17. | 0.0199 | 0.0155 | 0.0174 | 0.0151 | 0.0129 | 0.0130 | 0.0155 | 0.0136 |
| 18. | 0.0129 | 0.0129 | 0.0124 | 0.0128 | 0.0125 | 0.0129 | 0.0128 | 0.0128 |
| 19. | 0.0144 | 0.0133 | 0.0136 | 0.0142 | 0.0136 | 0.0133 | 0.0135 | 0.0132 |
| 20. | 0.0132 | 0.0132 | 0.0130 | 0.0131 | 0.0132 | 0.0132 | 0.0142 | 0.0173 |
| 21. | 0.0127 | 0.0126 | 0.0147 | 0.0186 | 0.0127 | 0.0184 | 0.0149 | 0.0130 |
| 22. | 0.0162 | 0.0149 | 0.0128 | 0.0129 | 0.0131 | 0.0129 | 0.0162 | 0.0151 |
| 23. | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0138 |
| 24. | 0.0159 | 0.0171 | 0.0126 | 0.0126 | 0.0190 | 0.0201 | 0.0126 | 0.0126 |
| 25. | 0.0125 | 0.0131 | 0.0130 | 0.0130 | 0.0131 | 0.0129 | 0.0132 | 0.0133 |
| 26. | 0.0129 | 0.0132 | 0.0130 | 0.0126 | 0.0135 | 0.0134 | 0.0132 | 0.0130 |
| 27. | 0.0131 | 0.0132 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0133 | 0.0131 |

Table 5.6. LCS for oily fingerprint image.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0129 | 0.0153 | 0.0129 | 0.0158 | 0.0141 | 0.0138 | 0.0128 | 0.0153 |
| 2. | 0.0130 | 0.0122 | 0.0127 | 0.0138 | 0.0131 | 0.0131 | 0.0144 | 0.0139 |
| 3. | 0.0131 | 0.0132 | 0.0130 | 0.0129 | 0.0130 | 0.0118 | 0.0128 | 0.0129 |
| 4. | 0.0131 | 0.0134 | 0.0131 | 0.0132 | 0.0130 | 0.0131 | 0.0131 | 0.0131 |
| 5. | 0.0131 | 0.0130 | 0.0145 | 0.0132 | 0.0132 | 0.0130 | 0.0132 | 0.0132 |
| 6. | 0.0130 | 0.0133 | 0.0130 | 0.0130 | 0.0129 | 0.0129 | 0.0131 | 0.0131 |
| 7. | 0.0122 | 0.0132 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 |
| 8. | 0.0131 | 0.0133 | 0.0130 | 0.0130 | 0.0129 | 0.0130 | 0.0130 | 0.0130 |
| 9. | 0.0129 | 0.0130 | 0.0129 | 0.0128 | 0.0127 | 0.0126 | 0.0125 | 0.0129 |
| 10. | 0.0128 | 0.0130 | 0.0129 | 0.0130 | 0.0122 | 0.0124 | 0.0129 | 0.0129 |
| 11. | 0.0130 | 0.0130 | 0.0129 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0139 |
| 12. | 0.0129 | 0.0131 | 0.0131 | 0.0130 | 0.0131 | 0.0130 | 0.0133 | 0.0129 |
| 13. | 0.0127 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0131 |
| 14. | 0.0129 | 0.0130 | 0.0135 | 0.0129 | 0.0130 | 0.0129 | 0.0130 | 0.0129 |
| 15. | 0.0104 | 0.0104 | 0.0111 | 0.0110 | 0.0108 | 0.0135 | 0.0125 | 0.0130 |
| 16. | 0.0130 | 0.0129 | 0.0131 | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0130 |
| 17. | 0.0115 | 0.0151 | 0.0149 | 0.0135 | 0.0126 | 0.0129 | 0.0123 | 0.0121 |
| 18. | 0.0130 | 0.0124 | 0.0130 | 0.0133 | 0.0128 | 0.0129 | 0.0126 | 0.0130 |
| 19. | 0.0121 | 0.0132 | 0.0129 | 0.0107 | 0.0151 | 0.0107 | 0.0139 | 0.0123 |
| 20. | 0.0102 | 0.0114 | 0.0103 | 0.0130 | 0.0122 | 0.0118 | 0.0110 | 0.0101 |
| 21. | 0.0103 | 0.0108 | 0.0102 | 0.0123 | 0.0130 | 0.0125 | 0.0121 | 0.0131 |
| 22. | 0.0107 | 0.0106 | 0.0109 | 0.0132 | 0.0117 | 0.0137 | 0.0146 | 0.0128 |
| 23. | 0.0133 | 0.0129 | 0.0131 | 0.0130 | 0.0130 | 0.0130 | 0.0129 | 0.0132 |
| 24. | 0.0128 | 0.0124 | 0.0111 | 0.0121 | 0.0128 | 0.0127 | 0.0120 | 0.0120 |
| 25. | 0.0124 | 0.0132 | 0.0123 | 0.0124 | 0.0120 | 0.0135 | 0.0134 | 0.0130 |
| 26. | 0.0126 | 0.0132 | 0.0129 | 0.0130 | 0.0131 | 0.0127 | 0.0131 | 0.0130 |
| 27. | 0.0130 | 0.0130 | 0.0130 | 0.0138 | 0.0130 | 0.0130 | 0.0130 | 0.0132 |
| 28. | 0.0127 | 0.0130 | 0.0129 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0129 |

## 5.2. The Ratio For Ridge Thickness To Valley Thickness Of Fingerprint Images

The ratio for ridge thickness to valley thickness is computed for each block. The image is divided into several blocks that do not overlap with the size of *MxN* pixels. Ridge thickness and valley thickness are obtained using gray level values for one image block in the direction normal to ridge flow. After that, the ratio of each block is computed and average value of the ratio is obtained over the whole image(see appendix A). Table 5.7 shows the ratio for ridge thickness to valley thickness for dry images. Table 5.8 shows the ratio for ridge thickness to valley thickness for neutral images.

Table 5.7. The ratio for ridge thickness to valley thickness for dry images.

| FINGER / Person | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 6.7557e-005 | 6.9074e-005 | 7.1427e-005 | 7.6393e-005 | 7.3589e-005 | 6.7169e-005 | 7.1224e-005 | 7.1282e-005 |
| 2. | 6.6096e-005 | 7.0580e-005 | 7.0274e-005 | 6.9991e-005 | 7.0595e-005 | 6.9732e-005 | 7.0501e-005 | 6.9274e-005 |
| 3. | 7.2351e-005 | 6.7538e-005 | 7.2062e-005 | 6.7047e-005 | 6.5351e-005 | 6.6496e-005 | 7.2768e-005 | 6.9722e-005 |
| 4. | 6.6639e-005 | 6.5760e-005 | 6.7272e-005 | 6.7639e-005 | 6.5061e-005 | 6.9698e-005 | 6.7198e-005 | 7.2071e-005 |
| 5. | 6.4641e-005 | 6.6624e-005 | 6.4758e-005 | 6.8665e-005 | 6.5816e-005 | 6.6588e-005 | 7.1107e-005 | 6.5322e-005 |
| 6. | 6.7217e-005 | 6.7685e-005 | 6.5558e-005 | 6.6158e-005 | 6.6808e-005 | 6.8684e-005 | 6.8603e-005 | 6.6808e-005 |
| 7. | 6.8431e-005 | 6.6019e-005 | 7.0428e-005 | 7.0746e-005 | 6.7458e-005 | 6.5847e-005 | 6.8724e-005 | 7.0724e-005 |
| 8. | 6.5428e-005 | 6.7265e-005 | 6.4173e-005 | 6.7124e-005 | 6.3950e-005 | 6.7965e-005 | 6.3070e-005 | 6.4673e-005 |
| 9. | 6.2755e-005 | 6.3926e-005 | 6.2754e-005 | 6.1443e-005 | 6.0622e-005 | 6.2608e-005 | 6.7797e-005 | 6.5276e-005 |
| 10. | 7.2156e-005 | 7.1226e-005 | 7.4735e-005 | 7.3667e-005 | 7.2057e-005 | 7.4825e-005 | 7.3282e-005 | 7.1493e-005 |
| 11. | 6.4636e-005 | 6.5400e-005 | 6.1447e-005 | 6.1805e-005 | 6.1808e-005 | 6.0367e-005 | 5.9686e-005 | 6.3777e-005 |
| 12. | 5.9396e-005 | 5.9823e-005 | 6.0736e-005 | 6.3255e-005 | 6.0329e-005 | 6.2011e-005 | 6.0812e-005 | 6.1229e-005 |
| 13. | 6.1436e-005 | 6.3353e-005 | 6.4445e-005 | 6.2834e-005 | 6.3667e-005 | 6.2906e-005 | 6.0277e-005 | 6.4344e-005 |
| 14. | 6.7841e-005 | 6.3055e-005 | 6.5662e-005 | 6.3961e-005 | 6.1443e-005 | 6.5306e-005 | 6.4931e-005 | 6.3895e-005 |
| 15. | 6.8155e-005 | 6.8420e-005 | 6.9555e-005 | 6.7186e-005 | 6.8567e-005 | 6.9338e-005 | 6.4410e-005 | 6.7673e-005 |
| 16. | 6.1143e-005 | 6.5210e-005 | 6.6227e-005 | 6.6154e-005 | 6.8071e-005 | 6.6047e-005 | 7.3056e-005 | 7.0756e-005 |
| 17. | 7.0863e-005 | 7.4834e-005 | 7.3417e-005 | 6.6623e-005 | 6.9624e-005 | 7.2894e-005 | 7.2639e-005 | 6.8400e-005 |
| 18. | 6.7569e-005 | 6.4868e-005 | 6.9766e-005 | 6.9352e-005 | 6.4292e-005 | 6.1036e-005 | 6.9348e-005 | 6.9931e-005 |
| 19. | 7.1828e-005 | 7.0986e-005 | 7.1534e-005 | 7.1916e-005 | 7.3568e-005 | 7.4425e-005 | 7.2997e-005 | 6.8629e-005 |
| 20. | 7.3475e-005 | 7.3475e-005 | 7.0314e-005 | 6.8189e-005 | 7.2177e-005 | 7.1937e-005 | 7.2687e-005 | 7.4595e-005 |
| 21. | 6.4367e-005 | 6.7008e-005 | 6.6667e-005 | 6.6489e-005 | 6.2617e-005 | 6.5449e-005 | 6.3243e-005 | 6.9932e-005 |
| 22. | 7.4322e-005 | 7.4278e-005 | 7.5352e-005 | 7.4601e-005 | 7.1108e-005 | 7.0907e-005 | 7.3319e-005 | 7.1931e-005 |
| 23. | 5.9744e-005 | 6.3776e-005 | 6.1024e-005 | 6.2936e-005 | 6.1383e-005 | 6.1790e-005 | 5.9693e-005 | 6.3812e-005 |
| 24. | 6.5392e-005 | 6.7352e-005 | 7.2311e-005 | 7.0793e-005 | 6.9560e-005 | 7.3167e-005 | 7.2789e-005 | 7.0186e-005 |
| 25. | 6.7899e-005 | 6.8649e-005 | 6.8413e-005 | 7.4574e-005 | 6.8235e-005 | 6.8345e-005 | 7.1091e-005 | 6.8876e-005 |

Table 5.8. The ratio for ridge thickness to valley thickness for neutral images.

| FINGER / Person | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 6.9037e-005 | 6.9613e-005 | 6.9779e-005 | 7.1505e-005 | 6.8312e-005 | 6.8731e-005 | 6.9732e-005 | 7.0054e-005 |
| 2. | 6.7974e-005 | 6.3096e-005 | 6.6922e-005 | 6.9151e-005 | 6.6895e-005 | 6.7987e-005 | 6.6807e-005 | 6.7285e-005 |
| 3. | 6.6985e-005 | 5.9779e-005 | 6.3770e-005 | 6.6843e-005 | 6.0812e-005 | 5.9939e-005 | 6.4536e-005 | 6.3459e-005 |
| 4. | 6.4860e-005 | 6.7721e-005 | 6.2999e-005 | 6.2821e-005 | 6.3998e-005 | 6.3225e-005 | 6.4744e-005 | 6.3695e-005 |
| 5. | 6.6191e-005 | 6.5079e-005 | 6.3057e-005 | 7.0652e-005 | 6.6317e-005 | 6.2335e-005 | 6.3038e-005 | 7.0346e-005 |
| 6. | 6.8606e-005 | 6.5672e-005 | 6.8709e-005 | 6.4341e-005 | 6.5472e-005 | 6.8140e-005 | 6.6426e-005 | 6.6697e-005 |
| 7. | 7.3957e-005 | 7.1498e-005 | 7.2182e-005 | 7.0347e-005 | 6.5796e-005 | 6.9691e-005 | 7.0394e-005 | 6.5338e-005 |
| 8. | 6.7923e-005 | 6.2572e-005 | 6.4817e-005 | 6.2264e-005 | 6.2734e-005 | 6.2546e-005 | 6.4443e-005 | 6.0645e-005 |
| 9. | 5.7197e-005 | 5.7194e-005 | 5.6957e-005 | 5.6906e-005 | 5.7039e-005 | 5.7149e-005 | 5.7628e-005 | 5.6981e-005 |
| 10. | 7.2485e-005 | 7.1425e-005 | 7.2449e-005 | 7.3565e-005 | 7.3135e-005 | 7.3098e-005 | 7.1771e-005 | 7.2919e-005 |
| 11. | 7.0703e-005 | 6.3531e-005 | 6.4254e-005 | 6.6870e-005 | 6.1439e-005 | 6.2656e-005 | 6.1279e-005 | 6.3679e-005 |
| 12. | 6.4494e-005 | 6.2804e-005 | 5.9651e-005 | 6.0465e-005 | 6.1821e-005 | 6.1686e-005 | 5.9498e-005 | 5.8230e-005 |
| 13. | 5.9051e-005 | 5.9800e-005 | 6.0544e-005 | 5.9761e-005 | 6.1043e-005 | 6.2609e-005 | 5.8358e-005 | 6.1991e-005 |
| 14. | 6.8390e-005 | 6.2293e-005 | 5.9511e-005 | 5.9032e-005 | 5.8306e-005 | 5.8856e-005 | 5.8578e-005 | 5.8658e-005 |
| 15. | 7.1030e-005 | 6.7342e-005 | 6.8032e-005 | 6.8465e-005 | 6.4230e-005 | 6.6969e-005 | 6.0966e-005 | 6.6653e-005 |
| 16. | 6.9808e-005 | 6.2900e-005 | 6.7400e-005 | 6.4108e-005 | 6.2099e-005 | 6.3062e-005 | 6.2427e-005 | 6.1203e-005 |
| 17. | 7.1492e-005 | 7.3784e-005 | 6.9280e-005 | 6.7444e-005 | 6.7626e-005 | 6.5536e-005 | 6.8969e-005 | 6.7051e-005 |
| 18. | 6.2305e-005 | 6.4313e-005 | 6.6668e-005 | 6.3304e-005 | 7.0897e-005 | 6.0996e-005 | 6.5376e-005 | 6.6423e-005 |
| 19. | 6.2617e-005 | 6.9489e-005 | 6.5787e-005 | 6.7875e-005 | 6.9308e-005 | 7.1455e-005 | 6.6778e-005 | 7.0626e-005 |
| 20. | 7.7842e-005 | 7.1989e-005 | 7.3784e-005 | 7.3393e-005 | 6.9860e-005 | 7.1476e-005 | 7.1592e-005 | 7.2837e-005 |
| 21. | 6.6631e-005 | 6.5886e-005 | 6.6895e-005 | 7.0835e-005 | 6.9196e-005 | 6.9541e-005 | 6.6637e-005 | 6.5512e-005 |
| 22. | 7.5196e-005 | 7.0723e-005 | 7.3297e-005 | 7.1395e-005 | 6.7828e-005 | 7.1688e-005 | 6.7068e-005 | 6.8255e-005 |
| 23. | 5.8975e-005 | 5.9089e-005 | 5.8817e-005 | 5.8070e-005 | 5.8141e-005 | 5.8257e-005 | 6.0221e-005 | 6.2906e-005 |
| 24. | 7.2007e-005 | 6.4899e-005 | 6.9139e-005 | 6.8549e-005 | 6.8970e-005 | 7.2787e-005 | 7.5253e-005 | 6.9438e-005 |
| 25. | 7.0194e-005 | 6.8216e-005 | 6.7841e-005 | 6.7564e-005 | 6.7651e-005 | 7.0230e-005 | 6.7968e-005 | 6.8357e-005 |

Table 5. 9. The ratio for ridge thickness to valley thickness for oily images

| FINGER / Person | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 6.2187e-005 | 6.6594e-005 | 6.2187e-005 | 6.2016e-005 | 6.3980e-005 | 6.5731e-005 | 6.2943e-005 | 6.3699e-005 |
| 2. | 6.3509e-005 | 6.4963e-005 | 6.3405e-005 | 6.2624e-005 | 6.2206e-005 | 6.1204e-005 | 6.4238e-005 | 6.2864e-005 |
| 3. | 5.9531e-005 | 6.2708e-005 | 6.0159e-005 | 6.1588e-005 | 5.8654e-005 | 6.4825e-005 | 6.4301e-005 | 6.0450e-005 |
| 4. | 6.1751e-005 | 6.2338e-005 | 6.4445e-005 | 6.0764e-005 | 6.1436e-005 | 6.1785e-005 | 6.1117e-005 | 6.0366e-005 |
| 5. | 6.1924e-005 | 6.2002e-005 | 6.2884e-005 | 6.1312e-005 | 6.3681e-005 | 5.9533e-005 | 6.2175e-005 | 5.9714e-005 |
| 6. | 5.9621e-005 | 6.3192e-005 | 6.2682e-005 | 6.2690e-005 | 6.1590e-005 | 6.0134e-005 | 6.4322e-005 | 6.6201e-005 |
| 7. | 6.5345e-005 | 6.1523e-005 | 5.8701e-005 | 5.8064e-005 | 5.8165e-005 | 5.7671e-005 | 5.8794e-005 | 5.9364e-005 |
| 8. | 6.3250e-005 | 6.2022e-005 | 6.1359e-005 | 6.1671e-005 | 6.5250e-005 | 6.2885e-005 | 6.5779e-005 | 6.6246e-005 |
| 9. | 5.8155e-005 | 5.7718e-005 | 5.8187e-005 | 5.8831e-005 | 6.0749e-005 | 6.0072e-005 | 5.9705e-005 | 5.9071e-005 |
| 10. | 6.5837e-005 | 6.1414e-005 | 6.1351e-005 | 5.8834e-005 | 6.3715e-005 | 6.3100e-005 | 6.1120e-005 | 6.0739e-005 |
| 11. | 5.6613e-005 | 5.7712e-005 | 5.8345e-005 | 6.1530e-005 | 5.9232e-005 | 6.1982e-005 | 6.3730e-005 | 6.0879e-005 |
| 12. | 5.6365e-005 | 5.8138e-005 | 5.6831e-005 | 5.6826e-005 | 5.6518e-005 | 5.6394e-005 | 5.9030e-005 | 5.7106e-005 |
| 13. | 5.9419e-005 | 5.7536e-005 | 5.6816e-005 | 5.6638e-005 | 5.6972e-005 | 5.7133e-005 | 5.7706e-005 | 5.7386e-005 |
| 14. | 6.3083e-005 | 6.1005e-005 | 6.0654e-005 | 6.1883e-005 | 6.0356e-005 | 6.0114e-005 | 6.0135e-005 | 6.0433e-005 |
| 15. | 7.4633e-005 | 7.3961e-005 | 7.2356e-005 | 7.0079e-005 | 7.1624e-005 | 6.8076e-005 | 6.4040e-005 | 6.0300e-005 |
| 16. | 5.9224e-005 | 6.1750e-005 | 6.2220e-005 | 5.9758e-005 | 5.9864e-005 | 6.0492e-005 | 6.3572e-005 | 5.8676e-005 |
| 17. | 6.6338e-005 | 6.4281e-005 | 6.1788e-005 | 6.1283e-005 | 6.4790e-005 | 6.0884e-005 | 6.4247e-005 | 6.4006e-005 |
| 18. | 6.0493e-005 | 6.1666e-005 | 5.9161e-005 | 5.9381e-005 | 5.8643e-005 | 5.8125e-005 | 6.0031e-005 | 5.9303e-005 |
| 19. | 7.3469e-005 | 6.5654e-005 | 6.5620e-005 | 7.5044e-005 | 7.1224e-005 | 7.3518e-005 | 7.0841e-005 | 7.0469e-005 |
| 20. | 7.3691e-005 | 6.8486e-005 | 7.3838e-005 | 6.6253e-005 | 6.6767e-005 | 7.0925e-005 | 7.1028e-005 | 7.5410e-005 |
| 21. | 7.5865e-005 | 7.2505e-005 | 7.6276e-005 | 7.0594e-005 | 6.8844e-005 | 6.7420e-005 | 6.8399e-005 | 6.8035e-005 |
| 22. | 7.3340e-005 | 7.3895e-005 | 7.2592e-005 | 6.5876e-005 | 7.0717e-005 | 6.5375e-005 | 6.5616e-005 | 6.5931e-005 |
| 23. | 6.3960e-005 | 6.2811e-005 | 6.2835e-005 | 6.3091e-005 | 6.5539e-005 | 6.4639e-005 | 6.3130e-005 | 6.2217e-005 |
| 24. | 6.1391e-005 | 6.5383e-005 | 6.9693e-005 | 6.6118e-005 | 6.1183e-005 | 6.0873e-005 | 6.6379e-005 | 6.4750e-005 |

## 5.3 **Global Contrast Factor (GCF)**

Contrast in image processing is usually defined as a ratio between the darkest and the brightest spots of an image. The newly introduced Global Contrast Factor (GCF) corresponds closer to the human perception of contrast. GCF uses contrasts at various resolution levels in order to compute overall contrast. The contrast of any (small) part of an image is called the local contrast. The global contrast is defined as the average local contrast of smaller image fractions. First compute the local contrast factors at various resolutions, and then build a weighted average based on human perception method which it can be approximated with the square root of the linear luminance, which is gamma corrected luminance using a gamma of 2.2 for standard displays (see appendix A).Table 5.10 shows the global contrast factor for neutral images.

Table 5.10. The global contrast factor for neutral images.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 10.6390 | 10.6211 | 10.4639 | 10.0405 | 10.6695 | 10.7485 | 10.5718 | 10.4435 |
| 2. | 10.9799 | 11.6504 | 11.1250 | 10.6793 | 11.0022 | 11.0331 | 11.1509 | 10.8239 |
| 3. | 11.5778 | 11.0979 | 11.2354 | 10.7261 | 10.7776 | 10.1344 | 11.0445 | 11.2691 |
| 4. | 11.3768 | 10.9347 | 11.6558 | 11.8149 | 11.4606 | 11.7306 | 11.5973 | 11.6009 |
| 5. | 11.1903 | 11.4180 | 11.6687 | 10.3453 | 11.4020 | 11.7210 | 11.5738 | 10.2130 |
| 6. | 10.8715 | 12.1684 | 11.0328 | 12.2493 | 11.3218 | 10.8789 | 11.1841 | 11.8618 |
| 7. | 9.7700 | 10.4755 | 10.4021 | 10.8611 | 11.7306 | 10.9788 | 10.8981 | 11.9365 |
| 8. | 10.7929 | 11.4312 | 11.2163 | 11.3609 | 11.3206 | 11.2756 | 11.3760 | 11.4960 |
| 9. | 10.7651 | 10.8462 | 9.7592 | 10.5116 | 10.5449 | 9.8290 | 11.0740 | 11.1024 |
| 10. | 9.6204 | 10.0073 | 9.9176 | 9.5327 | 9.4757 | 9.6244 | 10.0335 | 9.5442 |
| 11. | 10.4987 | 11.5934 | 11.4924 | 11.1131 | 11.7088 | 11.7736 | 12.5077 | 11.6926 |
| 12. | 11.2721 | 11.6607 | 11.9491 | 11.6417 | 11.2899 | 11.5961 | 11.2608 | 11.2990 |
| 13. | 11.0803 | 11.6791 | 11.5030 | 11.4568 | 11.7148 | 11.5641 | 10.9571 | 11.4624 |
| 14. | 10.8777 | 11.2657 | 10.7127 | 11.2000 | 9.5449 | 9.4382 | 9.5271 | 9.4111 |
| 15. | 10.4669 | 11.1039 | 10.5497 | 10.5118 | 11.2517 | 11.1744 | 11.5591 | 10.9008 |
| 16. | 10.0110 | 11.3980 | 10.6226 | 11.3277 | 11.4834 | 11.3658 | 11.3273 | 11.5527 |
| 17. | 9.8789 | 9.2732 | 10.3908 | 10.8599 | 10.6735 | 11.1789 | 10.4625 | 10.8270 |
| 18. | 11.9326 | 12.1327 | 11.7650 | 12.0691 | 10.3191 | 12.1509 | 11.5347 | 11.3028 |
| 19. | 9.5684 | 10.5976 | 10.6809 | 10.5822 | 10.3863 | 10.0661 | 10.5195 | 9.4834 |
| 20. | 8.5192 | 9.7235 | 9.4246 | 9.4390 | 10.0198 | 9.6924 | 10.0374 | 9.2601 |

Table 5.11. The global contrast factor for dry images.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 10.8909 | 10.6000 | 9.9507 | 8.6157 | 10.0419 | 10.2489 | 10.0785 | 10.2556 |
| 2. | 10.3194 | 10.4392 | 10.3576 | 10.4759 | 10.3921 | 10.6078 | 10.4114 | 10.7086 |
| 3. | 9.8998 | 10.0684 | 9.9692 | 10.3998 | 10.1656 | 10.9932 | 9.5940 | 10.1152 |
| 4. | 10.8721 | 10.0941 | 10.8423 | 10.7646 | 10.3747 | 10.4062 | 10.0120 | 9.7761 |
| 5. | 10.4976 | 10.8023 | 10.2852 | 10.5508 | 10.0732 | 10.9514 | 9.8990 | 10.1501 |
| 6. | 10.0977 | 10.8719 | 10.7689 | 10.4830 | 10.8141 | 10.4288 | 10.6711 | 10.8141 |
| 7. | 10.4341 | 10.7799 | 10.4597 | 10.4300 | 10.1264 | 10.5147 | 10.8478 | 10.3680 |
| 8. | 10.0303 | 10.4145 | 10.8862 | 10.3154 | 10.8768 | 10.1876 | 11.0537 | 10.6165 |
| 9. | 10.9728 | 10.5316 | 10.3845 | 10.4640 | 10.8859 | 10.1927 | 10.8151 | 10.4581 |
| 10. | 10.1151 | 10.0447 | 9.3459 | 9.4211 | 10.2356 | 9.0985 | 9.4752 | 10.1250 |
| 11. | 10.5539 | 10.5851 | 10.0318 | 10.8335 | 10.1276 | 10.8740 | 11.9084 | 10.5125 |
| 12. | 10.3385 | 10.4794 | 10.7087 | 10.6003 | 10.6517 | 10.9471 | 11.4820 | 10.7668 |
| 13. | 10.3381 | 10.3169 | 10.2262 | 10.3088 | 10.3544 | 10.4580 | 10.2960 | 10.0420 |
| 14. | 10.9901 | 10.6855 | 10.4621 | 10.5050 | 10.5447 | 10.5099 | 10.3944 | 10.7012 |
| 15. | 10.5755 | 10.3139 | 10.1326 | 10.0312 | 10.3217 | 10.4291 | 10.2022 | 10.7111 |
| 16. | 11.4838 | 11.0184 | 10.9763 | 9.1233 | 10.3267 | 10.9059 | 9.3310 | 10.1765 |
| 17. | 10.1833 | 9.2845 | 9.5333 | 10.1275 | 10.5987 | 9.6037 | 9.7958 | 10.5999 |
| 18. | 10.0510 | 10.7700 | 10.7234 | 10.5714 | 10.8579 | 10.8936 | 10.1094 | 10.7956 |
| 19. | 10.3295 | 10.3131 | 10.0768 | 9.7476 | 9.6281 | 9.3931 | 9.6842 | 10.4817 |
| 20. | 9.4012 | 10.0770 | 10.4968 | 9.8341 | 9.8426 | 9.8087 | 9.3791 | 9.0831 |
| 21. | 10.4346 | 10.8673 | 10.9819 | 10.0400 | 10.4965 | 10.3449 | 11.4018 | 10.1723 |
| 22. | 9.4069 | 9.6394 | 9.3554 | 9.4708 | 10.4410 | 10.2846 | 9.5871 | 10.2432 |
| 23. | 10.1078 | 10.1281 | 10.4516 | 10.4765 | 10.9693 | 10.1924 | 10.4488 | 10.3672 |
| 24. | 10.0539 | 10.0799 | 9.6989 | 10.1279 | 10.6708 | 9.5911 | 9.7511 | 10.3560 |
| 25. | 10.9007 | 10.6766 | 10.7413 | 9.2283 | 10.7915 | 10.7182 | 9.9875 | 10.6636 |
| 26. | 11.1313 | 10.1922 | 9.9946 | 9.6228 | 10.1718 | 10.4422 | 10.2253 | 10.5680 |
| 27. | 10.3328 | 10.5630 | 10.1654 | 10.0252 | 10.4972 | 9.6099 | 10.4326 | 10.4432 |
| 28. | 10.3433 | 10.6387 | 10.4569 | 10.0878 | 10.5220 | 10.0257 | 10.1928 | 10.9424 |
| 29. | 10.0277 | 9.3581 | 9.4725 | 10.1184 | 10.2990 | 10.5680 | 10.9628 | 9.9936 |
| 30. | 9.6233 | 9.8164 | 10.0180 | 10.3846 | 10.1965 | 9.8742 | 10.2701 | 10.8806 |
| 31. | 10.1448 | 10.0747 | 10.9238 | 9.7299 | 10.2555 | 10.7306 | 10.9872 | 10.5500 |
| 32. | 10.7179 | 9.3921 | 10.0465 | 9.3599 | 9.5552 | 10.5427 | 10.0830 | 10.2142 |

Table 5.12. The global contrast factor for oily images.

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 8.8731 | 8.0432 | 8.8731 | 9.1062 | 9.6690 | 9.4693 | 9.1112 | 9.4580 |
| 2. | 9.7102 | 9.7663 | 9.2221 | 9.3023 | 9.4871 | 9.3344 | 9.5642 | 9.3806 |
| 3. | 9.1863 | 9.1500 | 9.5007 | 9.3800 | 9.5095 | 9.3179 | 9.0991 | 9.3941 |
| 4. | 9.0492 | 9.4927 | 9.3038 | 9.7260 | 9.3645 | 9.5527 | 9.3402 | 9.7522 |
| 5. | 8.9457 | 8.7044 | 9.5859 | 9.4887 | 8.4590 | 10.0956 | 8.9676 | 9.5749 |
| 6. | 9.3100 | 8.9702 | 9.3109 | 9.7554 | 9.7326 | 8.7070 | 8.8463 | 8.1524 |
| 7. | 9.4839 | 9.5207 | 9.5214 | 9.3407 | 9.7917 | 9.7056 | 9.4569 | 9.3121 |
| 8. | 9.4562 | 9.8937 | 9.3447 | 9.1909 | 9.6389 | 9.6456 | 9.6587 | 9.3711 |
| 9. | 9.7015 | 9.2267 | 9.4045 | 9.0213 | 9.7878 | 9.4134 | 9.9440 | 9.0821 |
| 10. | 9.6244 | 9.9517 | 9.4136 | 9.3208 | 9.7855 | 9.3969 | 9.6940 | 9.4204 |
| 11. | 9.2379 | 9.3942 | 9.2297 | 9.5418 | 9.9987 | 9.2804 | 9.1404 | 9.0172 |
| 12. | 9.0446 | 9.7311 | 9.8107 | 9.1959 | 9.4493 | 9.0281 | 9.7015 | 9.6101 |
| 13. | 9.3290 | 9.1707 | 9.6259 | 9.7526 | 9.9486 | 9.7145 | 9.2677 | 9.3992 |
| 14. | 8.4995 | 8.6661 | 8.6752 | 8.4235 | 9.0314 | 8.9615 | 9.0039 | 8.8471 |
| 15. | 9.1373 | 9.8026 | 9.1764 | 9.1154 | 9.1224 | 9.5478 | 9.5448 | 9.6958 |
| 16. | 9.8896 | 9.0638 | 9.1756 | 9.8762 | 9.4890 | 9.7889 | 9.5589 | 9.7258 |
| 17. | 11.6605 | 9.3357 | 9.3229 | 9.7756 | 10.9932 | 11.5310 | 11.1101 | 11.6612 |
| 18. | 9.5179 | 10.1771 | 11.1575 | 11.9875 | 11.4535 | 10.5284 | 11.0697 | 9.8869 |
| 19. | 8.3098 | 8.3851 | 9.4622 | 8.1487 | 9.0571 | 8.8797 | 8.5845 | 9.4298 |
| 20. | 8.9450 | 9.9200 | 9.2126 | 9.7131 | 9.8773 | 9.2503 | 9.3094 | 9.2930 |
| 21. | 9.1522 | 9.1382 | 9.0139 | 9.7203 | 9.1028 | 9.0912 | 9.7979 | 9.2406 |
| 22. | 8.9008 | 8.5759 | 8.6442 | 9.4546 | 8.9795 | 9.2894 | 8.8327 | 9.5754 |
| 23. | 8.4393 | 8.7705 | 8.4477 | 8.3697 | 8.1782 | 8.1210 | 8.5770 | 8.6209 |
| 24. | 9.4162 | 9.9254 | 9.4092 | 9.0391 | 9.1887 | 9.6193 | 9.1386 | 9.7920 |
| 25. | 8.7899 | 9.1122 | 9.6466 | 9.8592 | 9.7010 | 8.9831 | 9.4058 | 9.0562 |
| 26. | 9.7498 | 9.4617 | 9.9629 | 9.6772 | 9.9866 | 9.1754 | 9.1204 | 9.8120 |
| 27. | 10.3639 | 10.0907 | 10.1090 | 9.7729 | 9.9192 | 10.6714 | 10.9297 | 10.5176 |
| 28. | 9.1454 | 9.3869 | 9.8795 | 9.0676 | 9.3035 | 8.8866 | 9.5176 | 8.7703 |
| 29. | 8.0735 | 9.5713 | 9.5299 | 9.9542 | 9.4582 | 9.6676 | 9.9698 | 9.6246 |
| 30. | 9.4650 | 9.5889 | 9.5950 | 9.4454 | 9.5111 | 9.0151 | 9.0775 | 9.1013 |
| 31. | 9.1258 | 9.3126 | 9.4475 | 9.6306 | 9.7424 | 9.7861 | 9.4552 | 9.3301 |
| 32. | 9.1572 | 8.7450 | 9.2560 | 8.8619 | 9.4715 | 8.7991 | 9.6340 | 9.3414 |
| 33. | 9.1460 | 8.9548 | 8.6838 | 8.6939 | 9.8484 | 9.3433 | 9.1969 | 10.9183 |

## 5.4 Quality Analysis -Fuzzy Inference System

Fingerprint image quality analysis has been developed and implemented using a fuzzy inference system (FIS). FIS has been developed using five stages. The stages are as follow:

### 5.4.1. The Fuzzy Inference System Editor

This stage shows the fuzzy inference system editor which displays the general information about the proposed method as show in figure 5.1, which shows the four inputs LCS, GCS, RVTR and GCF with their membership functions. It also shows the three different outputs oily, dry, and neutral.

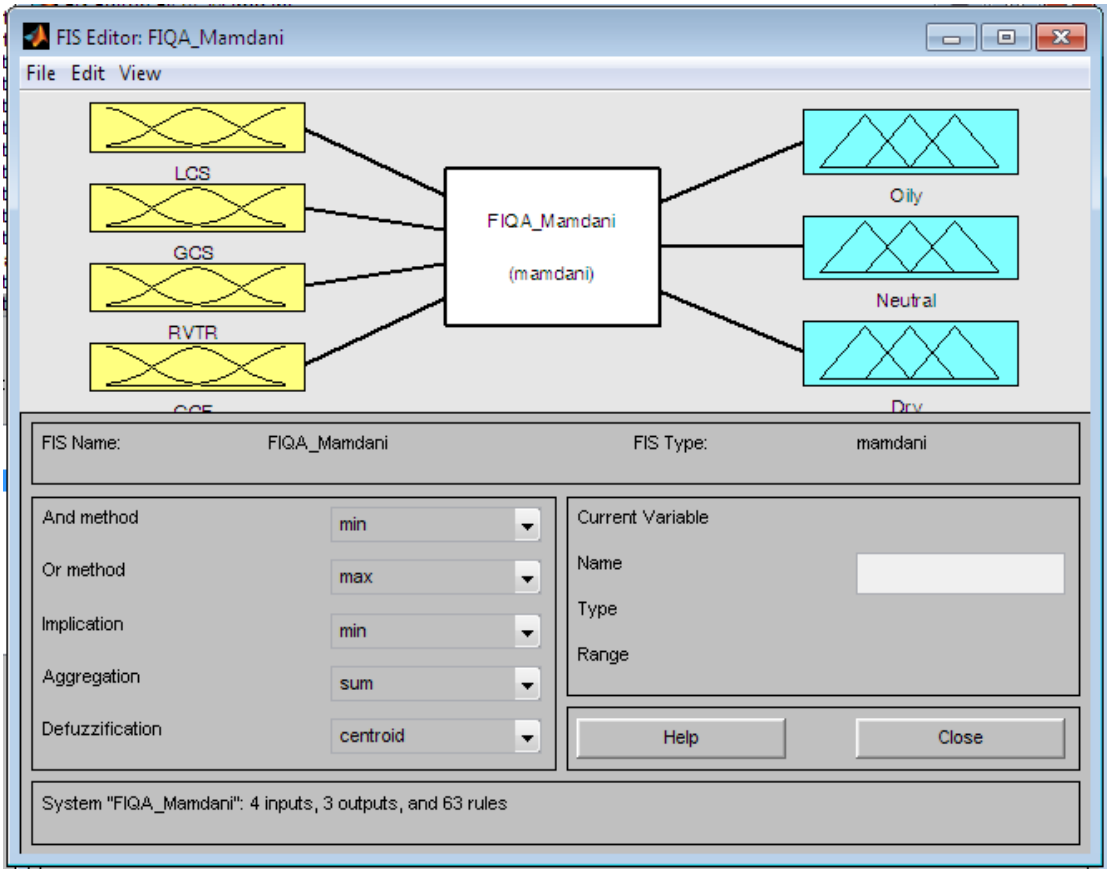


Figure 5.1.FIS Editor for image quality classification.

### 5.4.2. The Membership Function Editor

This stage shows the membership function for each of the four inputs. Figures 5.2, 5.3, 5.4, and 5.5 shows the membership function for GCF, LCS, GCS and RVTR, respectively.

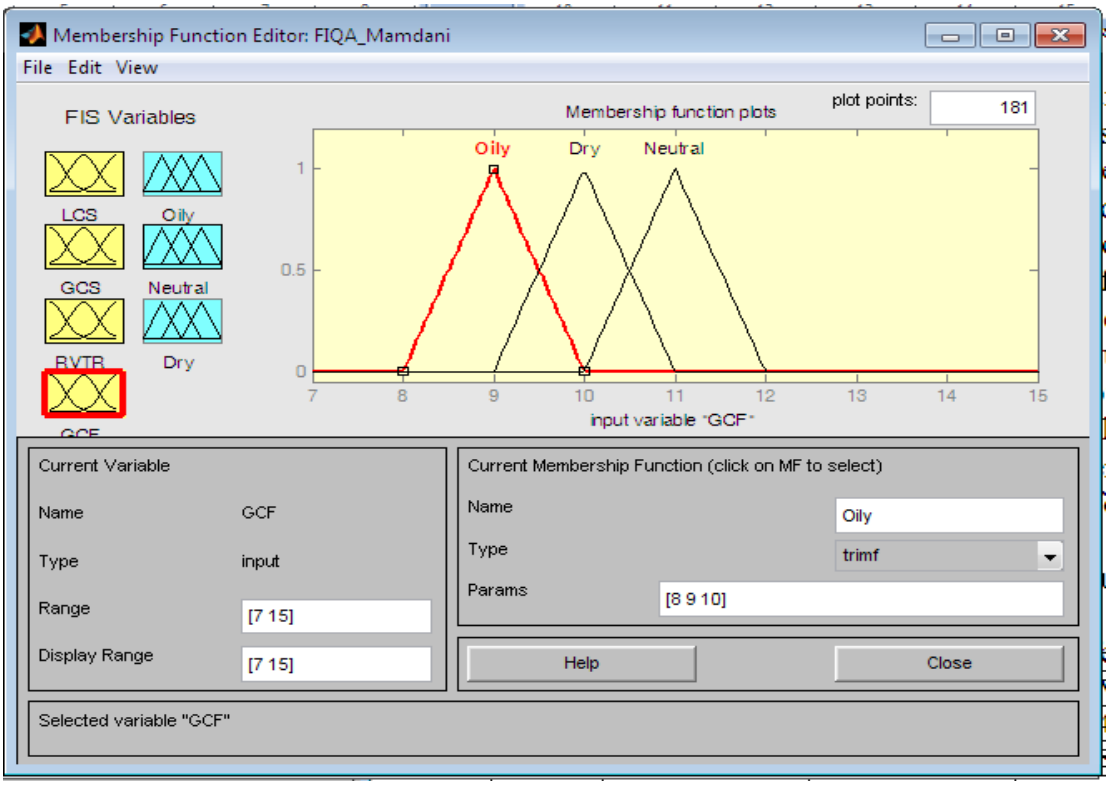


Figure 5.2.FIS Membership Function Editor for GCF.

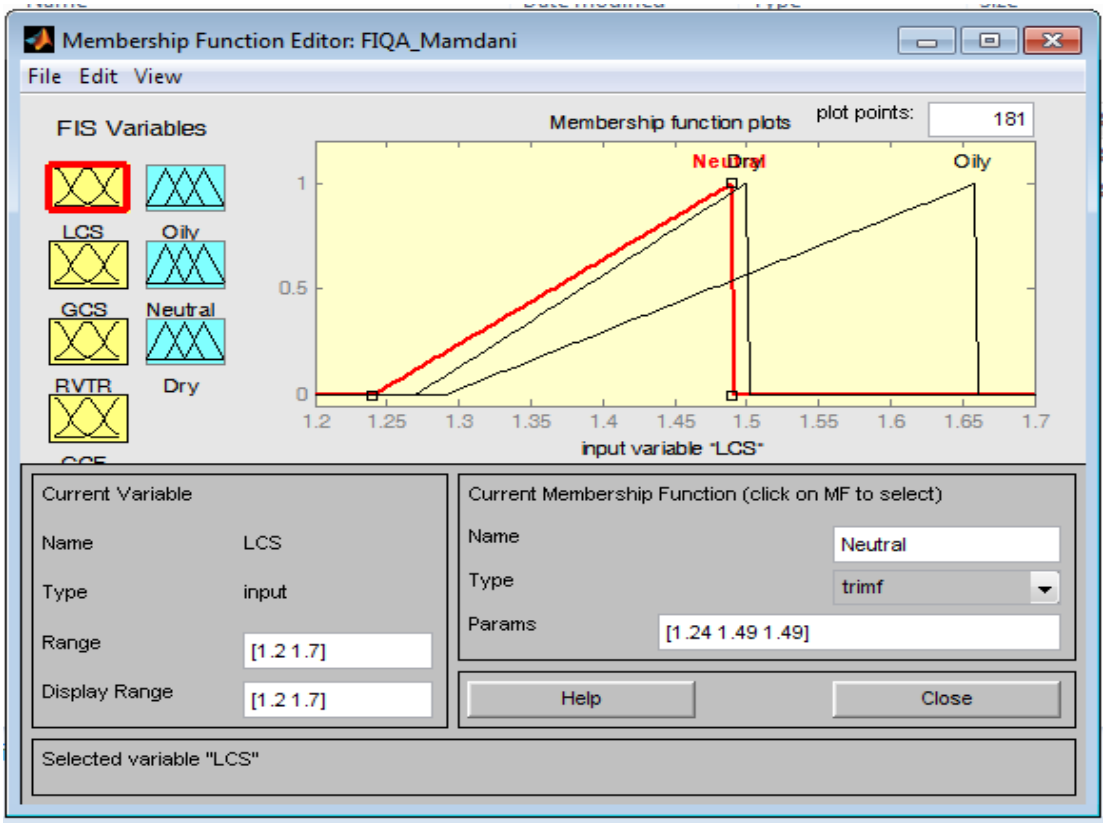


Figure 5.3. FIS Membership Function Editor for LCS.

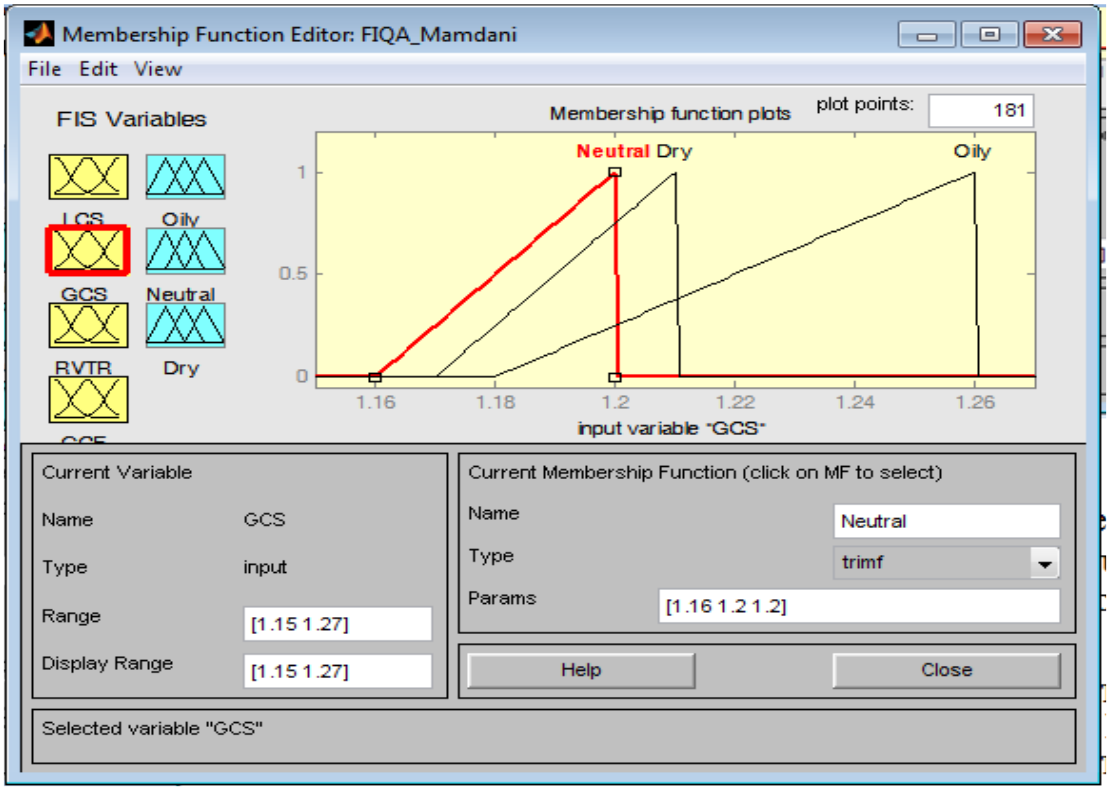


Figure 5.4. FIS Membership Function Editor for GCS.

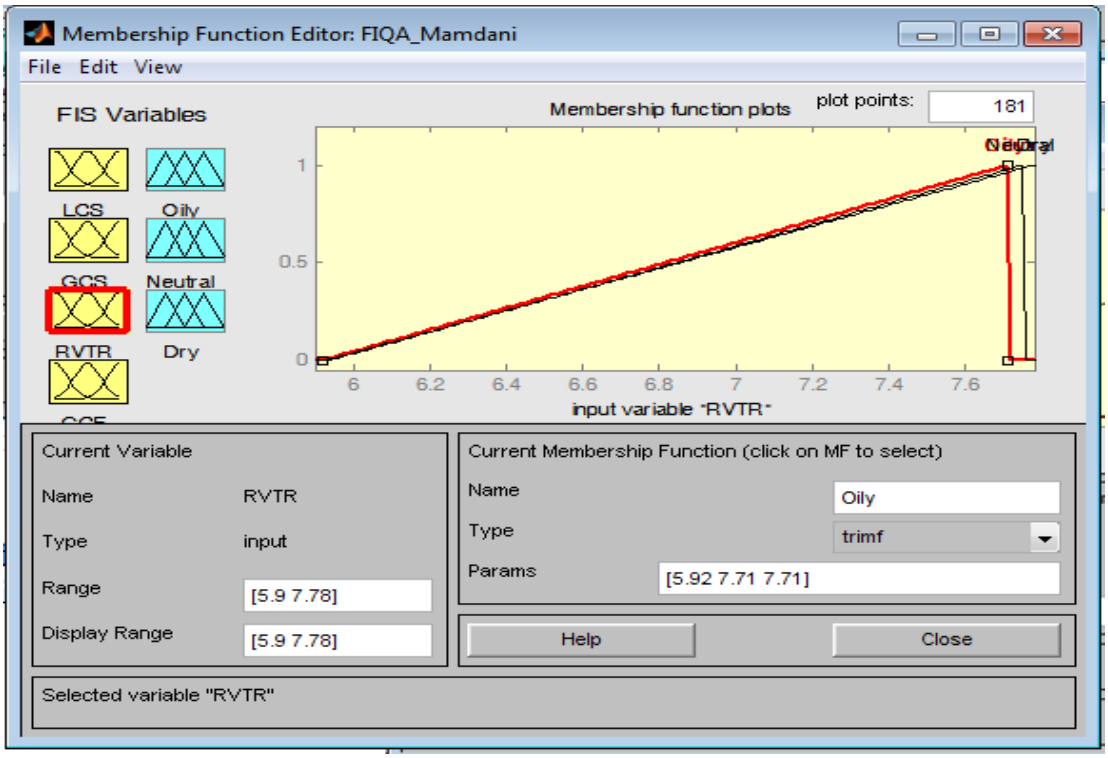


Figure 5.5. FIS Membership Function Editor for RVTR.

### 5.4.3. The Rules Editor

This stage shows the constructing rule base for four inputs and three outputs FIS. The FIS uses 63 rules which are defined as shown in table 5.13. Figure 5.6 shows the rule base editor.

Each output has two linguistic values. For example, the linguistic values for DRY fingerprints are S.Dry (Strong Dry) and M.Dry (Moderate Dry).

Table 5. 13. The FIS Rules.

| No | | LCS | | | GCS | | | RVTR | | | GCF | | | Image Type | Output |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. | IF | LCS | *Neutral* | and | GCS | *Neutral* | and | RVTR | *Neutral* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 2. | IF | LCS | *Neutral* | and | GCS | *Neutral* | and | RVTR | *Neutral* | and | GCF | *Oily* | Then | Image Quality | *Strong Neutral* |
| 3. | IF | LCS | *Neutral* | and | GCS | *Neutral* | and | RVTR | *Neutral* | and | GCF | *Dry* | Then | Image Quality | *Strong Neutral* |
| 4. | IF | LCS | *Neutral* | and | GCS | *Neutral* | and | RVTR | *Oily* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 5. | IF | LCS | *Neutral* | and | GCS | *Neutral* | and | RVTR | *Dry* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 6. | IF | LCS | *Neutral* | and | GCS | *Oily* | and | RVTR | *Neutral* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 7. | IF | LCS | *Neutral* | and | GCS | *Dry* | and | RVTR | *Neutral* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 8. | IF | LCS | *Oily* | and | GCS | *Neutral* | and | RVTR | *Neutral* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 9. | IF | LCS | *Dry* | and | GCS | *Neutral* | and | RVTR | *Neutral* | and | GCF | *Neutral* | Then | Image Quality | *Strong Neutral* |
| 10. | IF | LCS | *Neutral* | and | GCS | *Oily* | and | RVTR | *Oily* | and | GCF | *Oily* | Then | Image Quality | *Strong Oily* |
| 11. | IF | LCS | *Oily* | and | GCS | *Neutral* | and | RVTR | *Oily* | and | GCF | *Oily* | Then | Image Quality | *Strong Oily* |
| 12. | IF | LCS | *Oily* | and | GCS | *Oily* | and | RVTR | *Neutral* | and | GCF | *Oily* | Then | Image Quality | *Strong Oily* |
| 13. | IF | LCS | *Oily* | and | GCS | *Oily* | and | RVTR | *Oily* | and | GCF | *Neutral* | Then | Image Quality | *Strong Oily* |
| 14. | IF | LCS | *Oily* | and | GCS | *Oily* | and | RVTR | *Oily* | and | GCF | *Oily* | Then | Image Quality | *Strong Oily* |
| 15. | IF | LCS | *Oily* | and | GCS | *Oily* | and | RVTR | *Oily* | and | GCF | *Dry* | Then | Image Quality | *Strong Oily* |
| 16. | IF | LCS | *Oily* | and | GCS | *Oily* | and | RVTR | *Dry* | and | GCF | *Oily* | Then | Image Quality | *Strong Oily* |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **17.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Strong Oily*** |
| **18.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Strong Oily*** |
| **19.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **20.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **21.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **22.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **23.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***N*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **24.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **25.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***N*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **26.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **27.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Strong Dry*** |
| **28.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **29.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **30.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **31.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **32.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **33.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **34.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Neutral*** |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **35.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **36.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **37.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **38.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **39.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Neutral*** |
| **40.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **41.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **42.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **43.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **44.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **45.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **46.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **47.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **48.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **49.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **50.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **51.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Oily*** |
| **52.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Dry*** |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **53.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **54.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **55.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **56.** | **IF** | **LCS** | ***Neutral*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **57.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **58.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **59.** | **IF** | **LCS** | ***Oily*** | **and** | **GCS** | ***Dry*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **60.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Oily*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **61.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Neutral*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Oily*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **62.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Neutral*** | **and** | **GCF** | ***Dry*** | **Then** | **Image Quality** | ***Moderate Dry*** |
| **63.** | **IF** | **LCS** | ***Dry*** | **and** | **GCS** | ***Oily*** | **and** | **RVTR** | ***Dry*** | **and** | **GCF** | ***Neutral*** | **Then** | **Image Quality** | ***Moderate Dry*** |

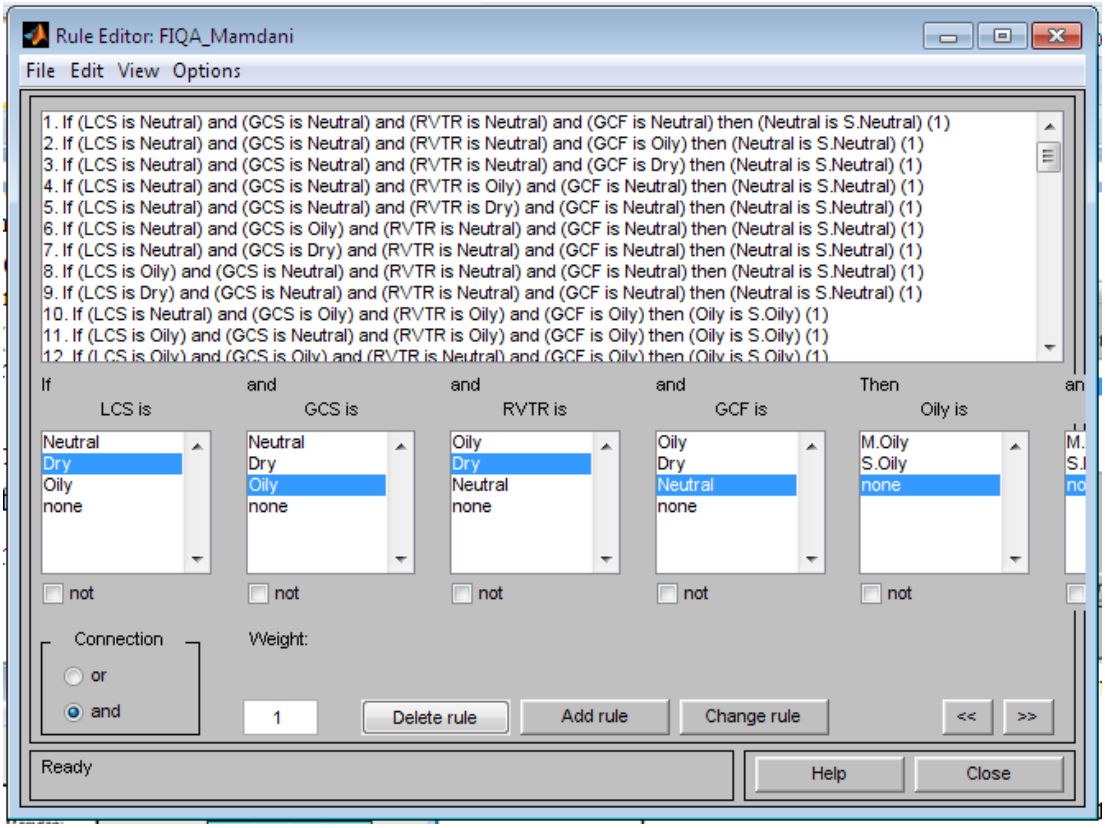


Figure 5.6. FIS Rule Editor.

We get the best analysis for fingerprint image by using FIS which uses five GUI tools for building, editing, and observing fuzzy inference systems. The tools are as follows:

- Fuzzy inference system editor
- Membership function editor
- Rule editor
- Rule viewer
- Surface viewer

The rules described in figure 5.6 were tested in our FIS with DB_ITS_2009 database [60]feature values. Tables 5.14, 5.15 and 5.16, as well as figures 5.7, 5.8, and 5.9 show the performance of our FIS, which determined the fingerprint image quality according to the features extracted.

Table 5.14. Oily fingerprint image.

| | | |
|---|---|---|
| **LCS** | **1.58** | **OILY** |
| **GCS** | **1.21** | |
| **RVTR** | **6.95** | |
| **GCF** | **9** | |

Table 5.15.Neutral fingerprint image.

| | | |
|---|---|---|
| **LCS** | **1.39** | **NEUTRAL** |
| **GCS** | **1.17** | |
| **RVTR** | **6.66** | |
| **GCF** | **11.1** | |

Table 5.16. Dry fingerprint image.

| | | |
|---|---|---|
| **LCS** | **1.41** | **DRY** |
| **GCS** | **1.19** | |
| **RVTR** | **7.78** | |
| **GCF** | **10** | |

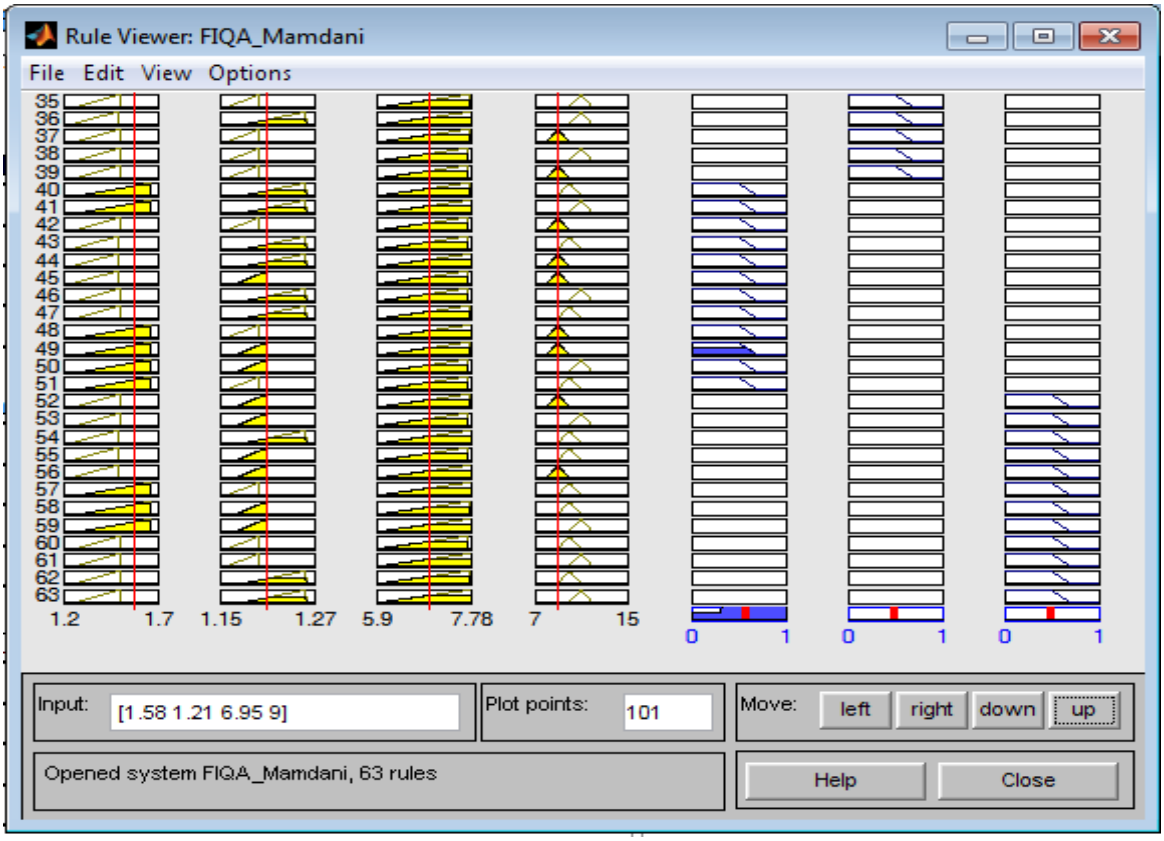


Figure 5.7. Rule viewer for oily fingerprint image.

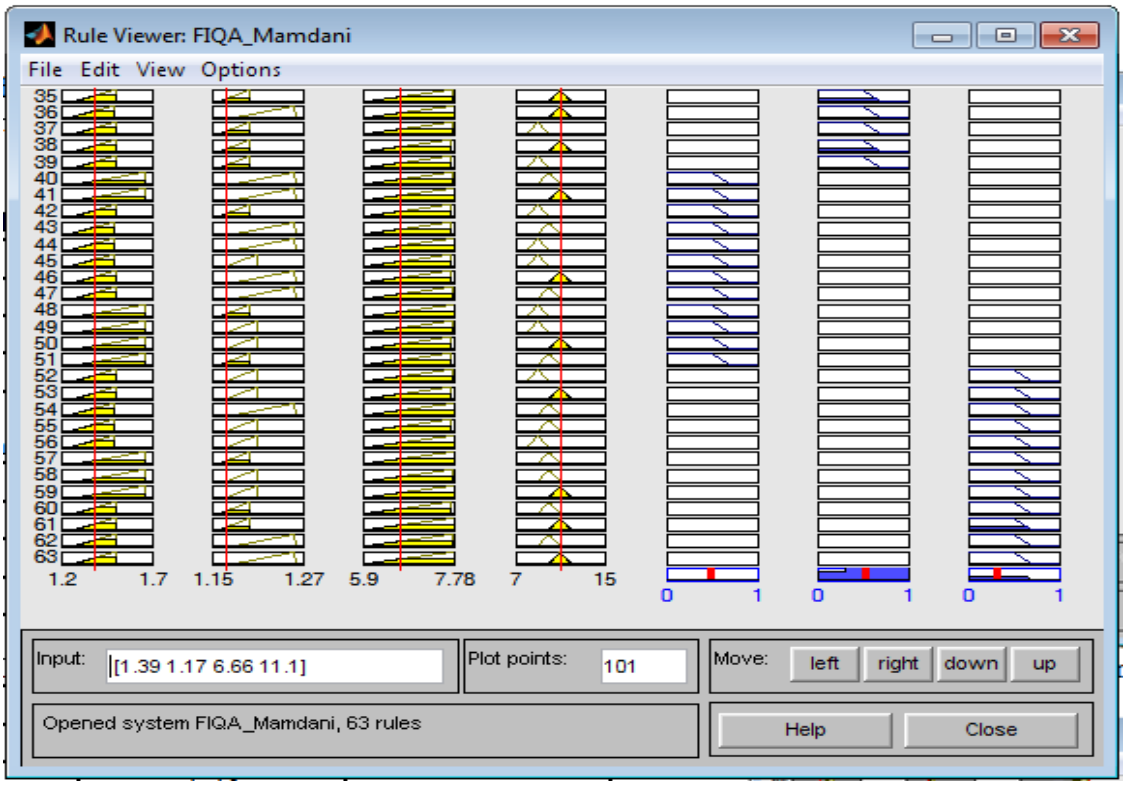


Figure 5.8. Rule viewer for neutral fingerprint image.

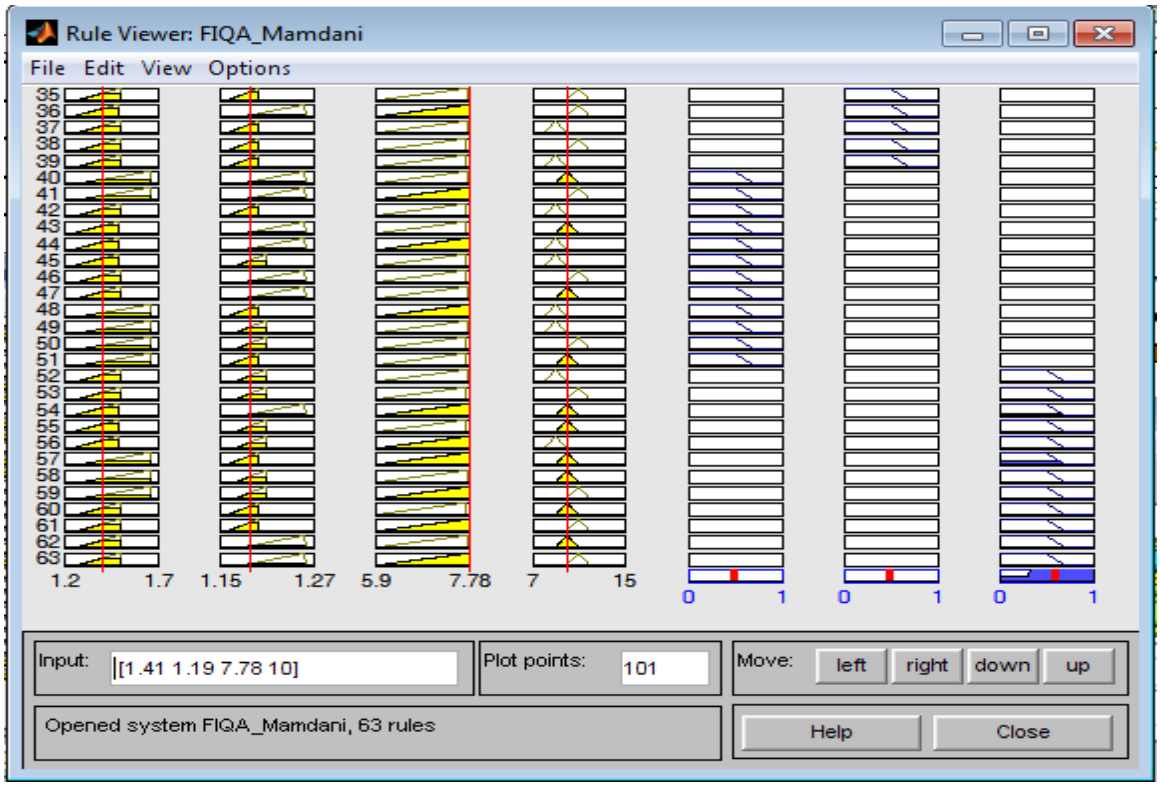


Figure 5.9.Rule viewer for dry fingerprint image.

The rule viewer, which display a roadmap of the whole fuzzy inference process, shows the values of input variables and their outputs, and has ability to change inputs and see output changes. Figures 5.7, 5.8, and 5.9 shows the rule viewer for Neutral, Oily and Dry fingerprint image, respectively for sample inputs.

The surface viewer presents three dimensional curve that represent two input features and the image quality type. Figures 5.10, 5.11, and 5.12 shows the three surface viewer for Dry, Neutral and Oily fingerprint image, respectively, which presents the two important features GCS and RVTR with an image quality type.

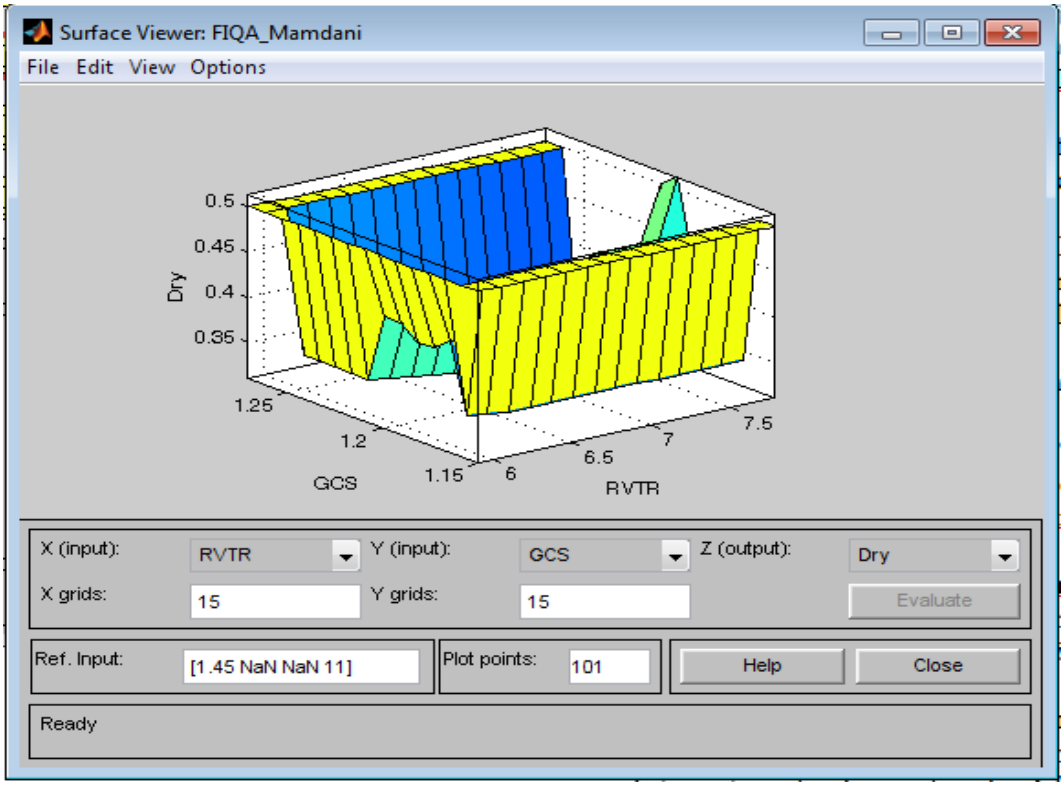


Figure 5.10.Surface viewer for Dry fingerprint image.

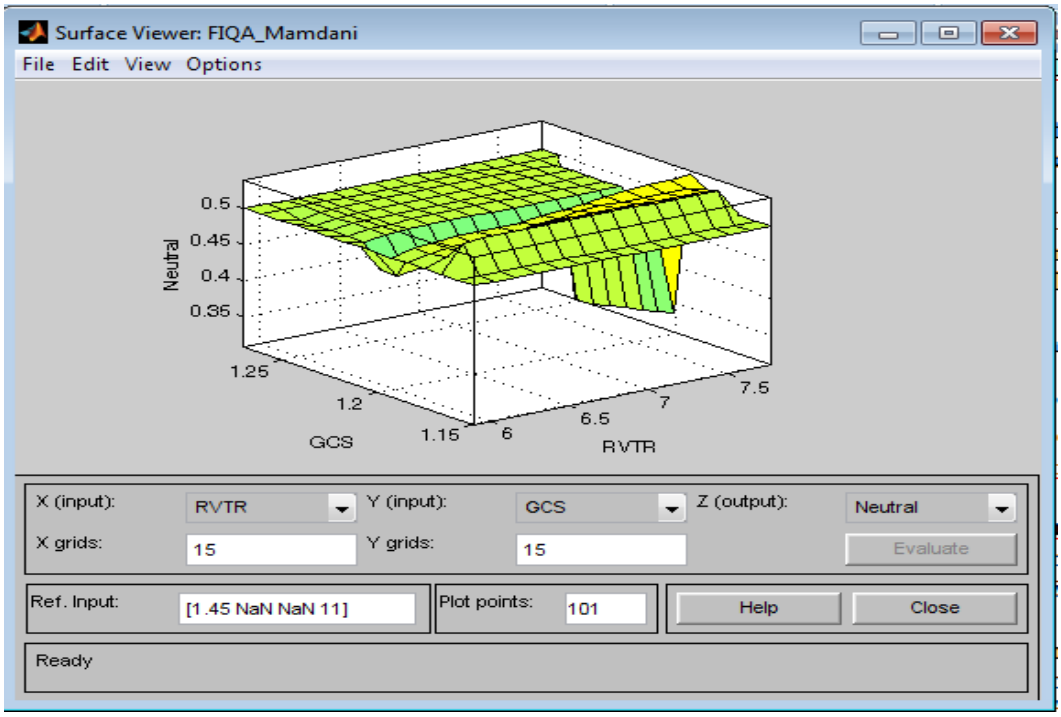


Figure 5.11.Surface viewer for Neutral fingerprint image.

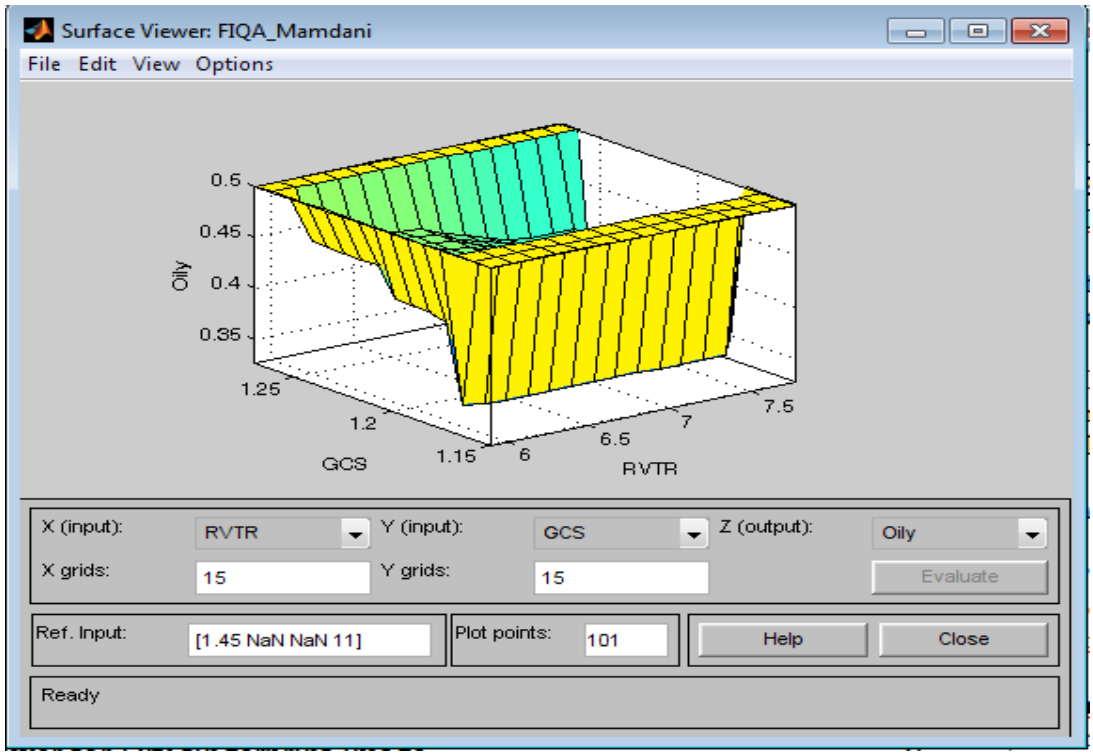


Figure 5.12.Surface viewer for Neutral fingerprint image.

## 5.5 Performance evaluation

To measure the performance of the proposed method used to analysis fingerprint image quality, we inputted the extracted features values for four features extraction (LCS, GCS, RVTR, and GCF) to the fuzzy inference system which it determined the type of fingerprint image according to their features values as we discussed above. The input operation it done for the four features as input, where the system give the output the type of the fingerprint image. Figure 5.13 shows the numbers of detected, and not detected for each type of fingerprint images. For oily fingerprint images the detected number is311 images, and not detected is257 images. For neutral fingerprint images the detected number is 389 images, and not detected is 179 images. For dry fingerprint image the detected is 462 images, and not detected is 106 images.

## 5.6 Theorem

The features LCS, GCS, RVTR, and GCF are enough to classify correctly fingerprint image to one of three types oily, dry or neutral.

The proof is by simulation as seen from the simulation presented in figure 5.13.Figure 5.13 shows numbers of detected and not detected for each type of fingerprint image, where the percentage of detected for each type as follow. For dry fingerprint the percentage is 81.33, for oily the percentage is 54.75, and for neutral the percentage is 68.48.

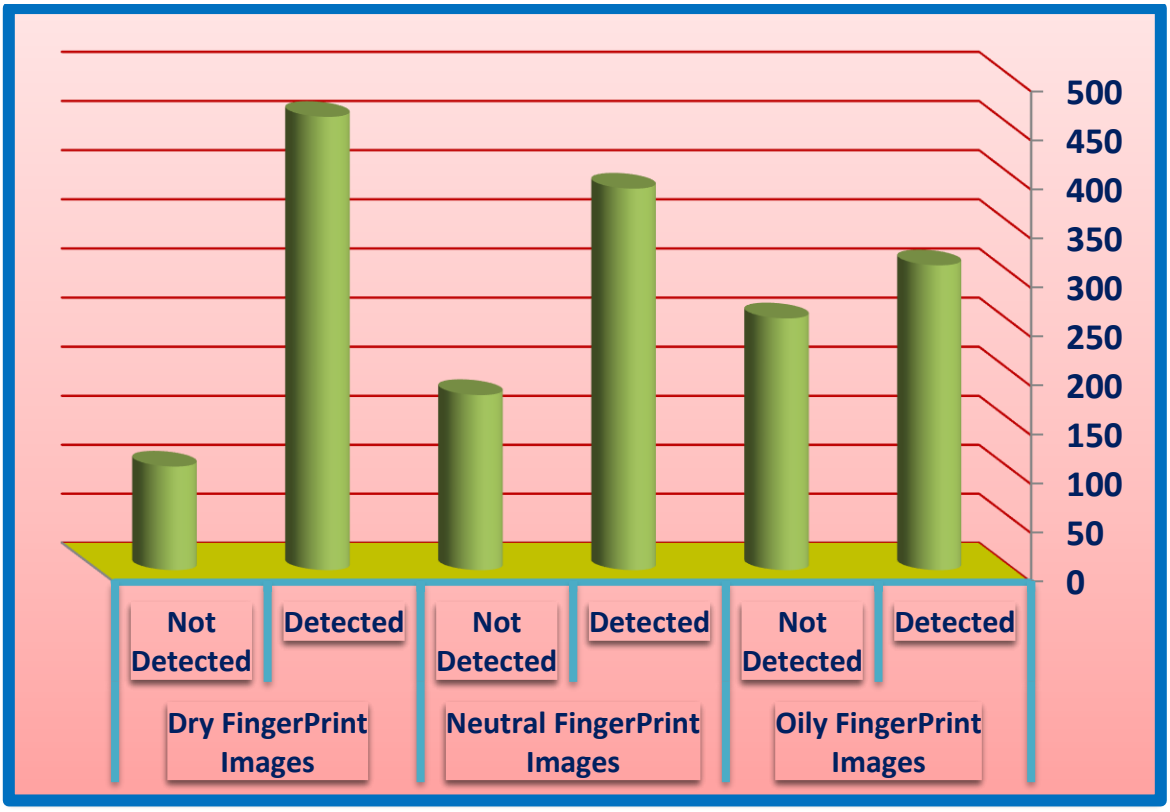


Figure 5.13. Fingerprint image Quality analysis

## 5.7 Henna fingerprint image (Sudanese case study )

Henna refers to the dye prepared from a plant and the temporary body art (tattooing or staining) based on those dyes. The art of applying henna to the hands and feet is traditionally used during social and religious celebrations. Every culture and region of the world uses henna tattoos in its own unique way. For Sudanese weddings, henna is painted on the bride and woman's to symbolize joy and beauty.

In this dissertation we proposed to study the effect of henna on fingerprint images. Optical sensor U.are.U 5000 USB Fingerprint Reader with the specifications: 512 dpi, USB 2.0, flat fingerprint, uncompressed to read fingerprint image but the device was not able to read the henna fingerprint images. The study propose to read fingerprint image before henna and after henna and then analyze the quality to determine the degree of effect of henna on fingerprint images, but the device used was, unfortunately, unable to read the henna fingerprint images.

## 5.8 Summary

In this chapter, we analyze fingerprint image quality by extracted four features LCS, GCS, RVTR and GCF form three types of fingerprint image (dry, Neutral and Oily) then used fuzzy inference system with this features. The fuzzy inference system using these four input features to analyze fingerprint image quality was designed. The FIS has four input variables and three output variables with sixty three rules.

# CHAPTER VI

## Dry Fingerprint Image Enhancement

Fingerprint identification is one of the most popular biometric technology and is used in criminal investigations, commercial applications, etc. The performance of a fingerprint image-matching algorithm depends heavily on the quality of the input fingerprint images. It is very important to acquire good quality images but in practice a significant percentage of acquired images are of poor quality due to some environmental factors such as temperature, humidity and pressure. The overall quality of the fingerprint depends greatly on the condition of the skin. Dry skin tends to cause inconsistent contact of the finger ridges with the scanner's platen surface, causing broken ridges and many white pixels replacing ridge structure. Therefore, enhancement of the quality and validity of the dry fingerprint image is necessary and meaningful. In this chapter we explain the implementation of dry fingerprint image enhancement and the result of implementation. Finally we measure the result by using FSIM.

### 6.1. The Method

The method used to enhancement dry fingerprint image is divided into three stages as follows:

1. Smoothing the inputted image using low pass filter.
2. Employ fuzzy morphology on the smoothed image.
3. The union operation is utilized to display results

The performance evolution is conducted to measure the quality of the enhanced image compared with previous studies using Features Similarity index metric (FSIM). Figure 6.1 shows the flowchart for the new proposed method used to enhance dry fingerprint image.

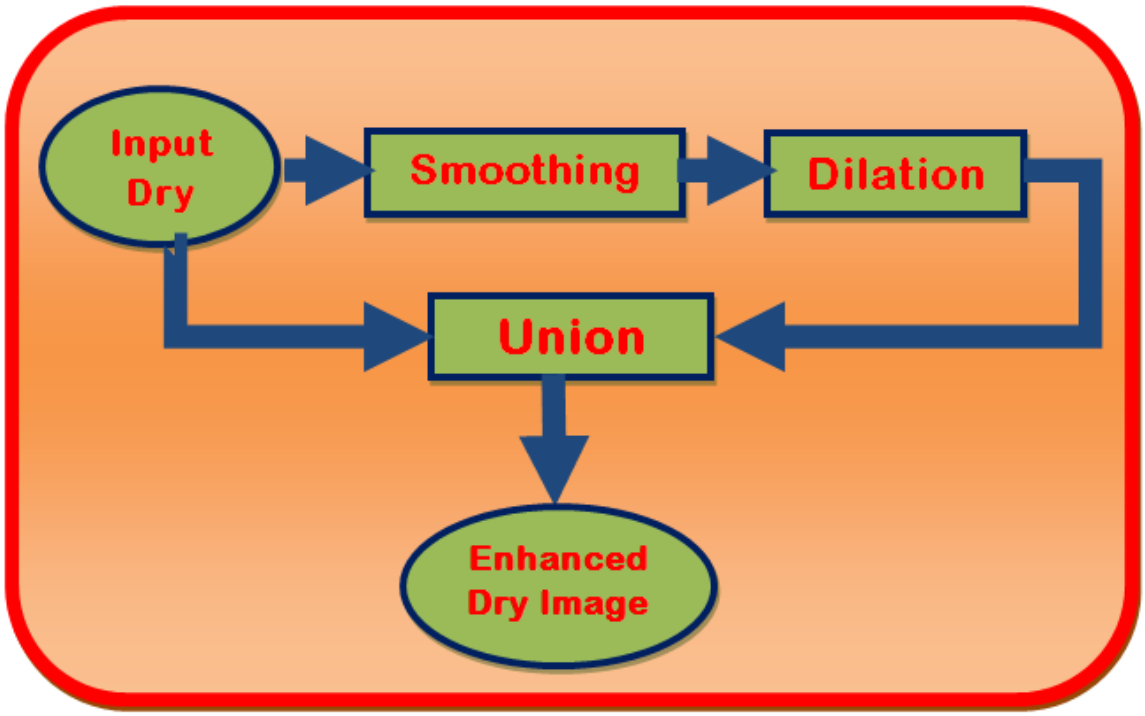


Figure 6.1. The flowchart for the proposed method.

In the first step, smoothing is applied to reduce noise in an image and to decrease the disparity between pixel values by using Winner filter. Figure 6.2 shows a smoothed fingerprint image.

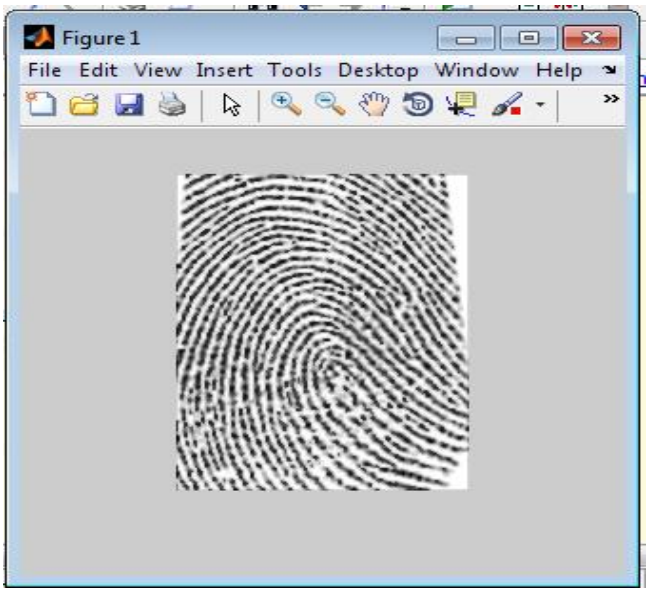


Figure 6.2. A smoothed image.

Fuzzy dilation is applied on the smoothed fingerprint image in stages. The first stage is to fuzzify the input image. The second stage is to apply the α-cut dilation and the third stage is to defuzzify which is the inverse of fuzzification. For image fuzzification we are required to employ a membership function where the membership values lie between 0 and 1. The selection of the membership function depends on the application. In image processing, heuristic membership functions are widely used to define certain properties (such as lightness or darkness of a pixel value). S-function is one of the most prominent heuristic membership function. The shape of S-function depends on three parameters; a, b, and c. The parameters a, b and c are specified to ensure the membership function maximizes the information contained in the image. In this study we will apply the S-membership function as shown below:

$$\begin{aligned} \mu(x) &= 0 && x<a, \\ &= 2\,[(x-a)/(c-a)]^2 && a < x < b, \\ &= 1 - 2\,[(x-c)/(c-a)]^2 && b < x < c, \\ &= 1 && x > c, \end{aligned}$$

where, b is any value between a and c. Figure 6.3 shows the membership function plot for a = Xmin, c = Xmax, for a normalized set.

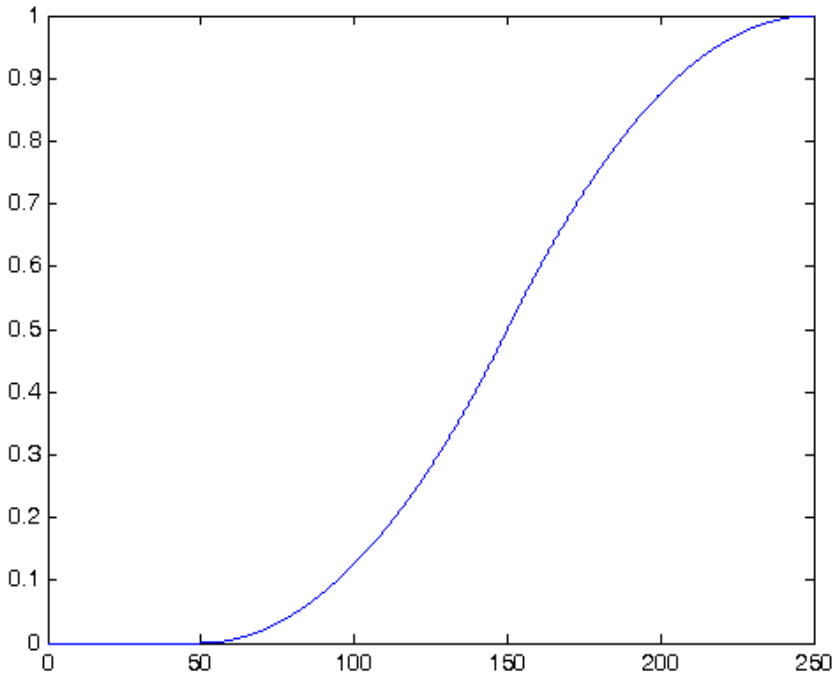


Figure 6.3. An S-membership function.

The fuzzified image can be created from the *S*-membership function with improved quality. The membership function $\mu = 1$ represents the maximum brightness. This study used the *S*-membership function to improve the quality of image.

Fuzzy dilation is applied by probe the structuring element to scan the whole image, and replace the center pixel in each probe of structuring element as in table 6.1 with pixel that satisfy the following formula:

$$[g(x) \oplus \mu(x)] \quad \alpha(x)=supmin[g(x-y),\mu(x)]$$

Where y∈X for Dilation.

Table 6.1. Structuring Element Mask

| | | |
|---|---|---|
| **0.3** | **0.8** | **0.3** |
| **0.8** | **1** | **0.8** |
| **0.3** | **0.8** | **0.3** |

Figure 6.4 shows an example of fuzzy dilated image.

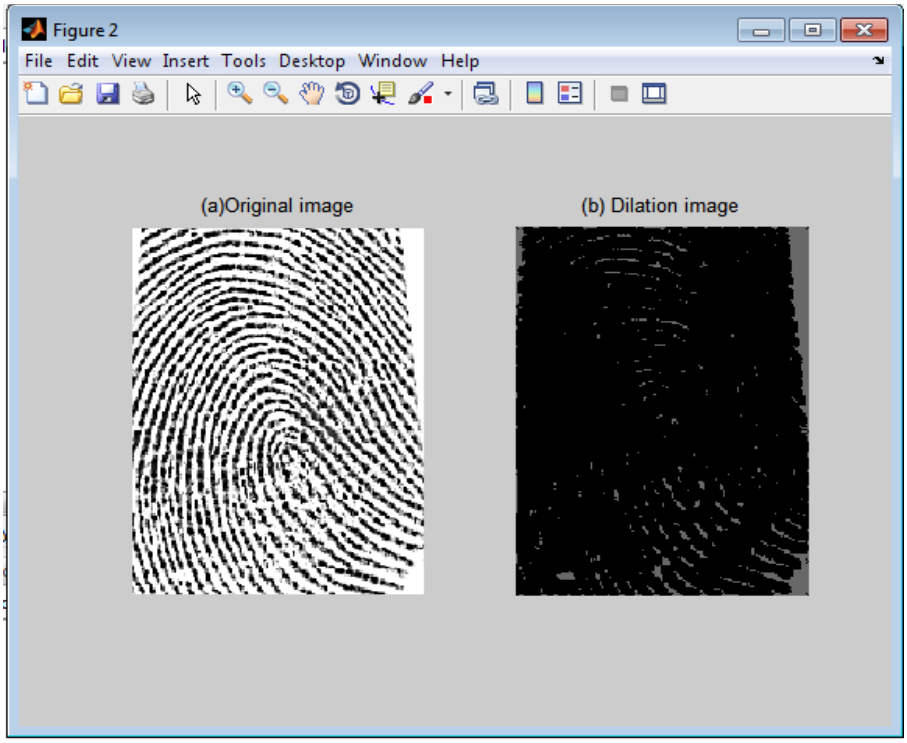


Figure 6.4. Fuzzy dilated image

The final step, the union operation step, where a logical operation exist that extracts the union of black pixels in an original image (input dry image) and the dilated image. In this step the white pixels in the ridges are removed and we obtain enhanced image with good ridges as shown in figure 6.5.

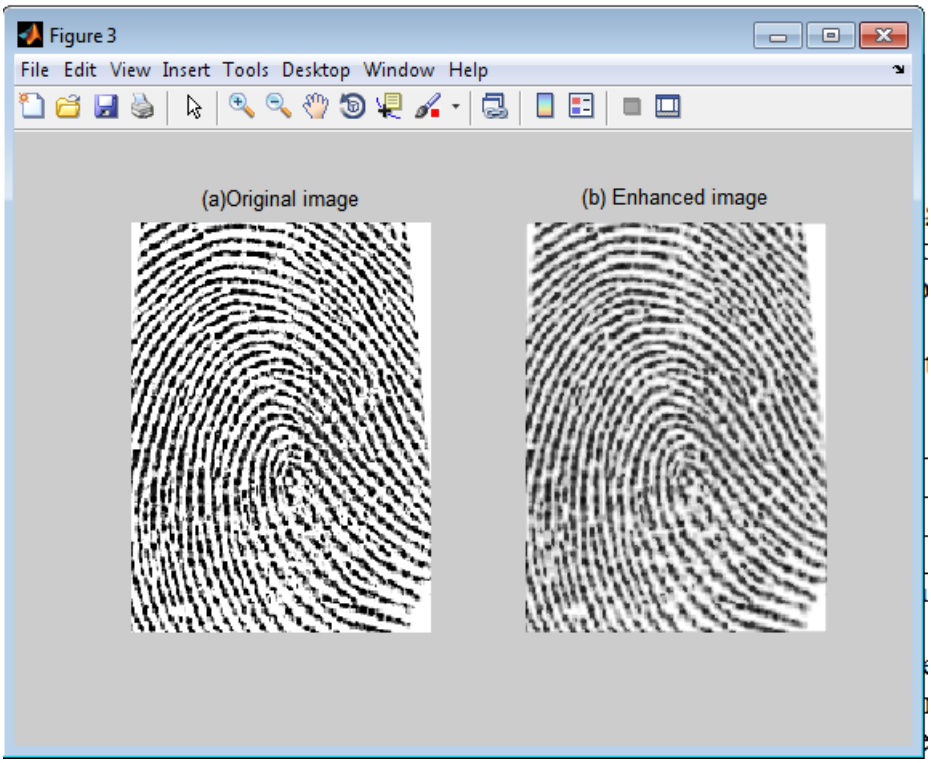


Figure 6.5. An Enhanced image.

The implementation of the proposed method was done using MATLAB and verified using the DB_ITS_2009 database[60](see appendix B), which is a private database collected by the Department of Electrical Engineering, Institute of Technology Sepuluh Nopember Surabaya. The database was taken with great caution because of the image quality considerations. The DB_ITS_2009 database was taken using an optical sensor U.are.U 4000B fingerprint reader with the specifications: 512 dpi, USB 2.0, flat fingerprint and uncompressed. This database has 1704 fingerprint images of size 154x208 pixels. The details are as follows: The fingerprint images are classified into three types the finger conditions (dry, neutral and oily). Each type of finger condition consists of 568 fingerprint images sourced from 71 different fingers. Each of these fingerprint images was taken eight times for the three conditions above. As a result, we obtained 3x71x8=1074 fingerprint images. To obtain dry fingerprint images, hair-dryer was used to completely dry the

fingertip. Likewise, in order to get oily fingerprint images, we smeared baby-oil on the fingertips before the image was taken [60].

Image smoothing is done using Winner filter, then fuzzy morphology operations are performed. A greyscale image is fuzzified with the use of the S-fuzzy membership functions. Then a fuzzy structuring element is traversed on the whole image to process the dilation operation. Figure 6.5 shows an example of an enhanced image through the proposed method.

We applied the feature similarity (FSIM) index is full reference image quality assessment (IQA) it base on feature rather than pixels exactly the low level features as human visual system (HVS) understand an image. The features are:

1. The phase congruency (PC) which is a dimensionless measure of the significance of a local structure, is used as the primary feature in FSIM.
2. Gradient magnitude (GM) features which represent complementary aspects of the image visual quality.

The computation of FSIM index consists of two stages. In the first stage, the local similarity map is computed, and then in the second stage, we pool the similarity map into a single similarity score, as in the following equation.

$$FSIM = \frac{\sum_{x \in \Omega} S_L(x).PC_m(X)}{\sum_{x \in \Omega} PC_m(X)} \quad 6.1$$

Where

Ωmeans the whole image spatial domain.

$S_L$ is the similarity between two images

PC is Phase congruency

The proposed dry fingerprint enhancement method using fuzzy morphology gives high FSIM when compared to the existing enhancement method by Eun-Kyung Yun et al. [6](see appendix D) that uses adaptive pre-processing method based on binary morphology image processing. The FSIM for the existing method and the proposed method are described in tables 6.2 and 6.3, respectively.

Table 6.2. The FSIM values for Eun-Kyung Yun et al. Method [6]

| Finger# Person # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0687 | 0.0641 | 0.0660 | 0.0601 | 0.0681 | 0.0693 | 0.0654 | 0.0818 |
| 2. | 0.0724 | 0.0696 | 0.0719 | 0.0723 | 0.0720 | 0.0692 | 0.0695 | 0.0731 |
| 3. | 0.0466 | 0.0600 | 0.0512 | 0.0604 | 0.0613 | 0.0593 | 0.0470 | 0.0518 |
| 4. | 0.0640 | 0.0613 | 0.0599 | 0.0594 | 0.0665 | 0.0543 | 0.0670 | 0.0541 |
| 5. | 0.0770 | 0.0748 | 0.0730 | 0.0614 | 0.0728 | 0.0704 | 0.0604 | 0.0714 |
| 6. | 0.0747 | 0.0738 | 0.0782 | 0.0747 | 0.0679 | 0.0647 | 0.0684 | 0.0679 |
| 7. | 0.0606 | 0.0610 | 0.0523 | 0.0524 | 0.0608 | 0.0649 | 0.0618 | 0.0580 |
| 8. | 0.0598 | 0.0597 | 0.0557 | 0.0511 | 0.0472 | 0.0504 | 0.0524 | 0.0476 |
| 9. | 0.0621 | 0.0606 | 0.0642 | 0.0647 | 0.0674 | 0.0638 | 0.0603 | 0.0621 |
| 10. | 0.0582 | 0.0540 | 0.0546 | 0.0603 | 0.0556 | 0.0566 | 0.0580 | 0.0599 |
| 11. | 0.0814 | 0.0794 | 0.0833 | 0.0791 | 0.0782 | 0.0772 | 0.0759 | 0.0796 |
| 12. | 0.0906 | 0.0678 | 0.0918 | 0.0899 | 0.0670 | 0.0849 | 0.0636 | 0.0904 |
| 13. | 0.0602 | 0.0593 | 0.0627 | 0.0564 | 0.0591 | 0.0597 | 0.0675 | 0.0603 |
| 14. | 0.0728 | 0.0654 | 0.0690 | 0.0714 | 0.0702 | 0.0706 | 0.0678 | 0.0747 |
| 15. | 0.0654 | 0.0732 | 0.0797 | 0.0690 | 0.0763 | 0.0805 | 0.0697 | 0.0710 |
| 16. | 0.0649 | 0.0760 | 0.0648 | 0.0641 | 0.0684 | 0.0696 | 0.0661 | 0.0628 |
| 17. | 0.0670 | 0.0633 | 0.0608 | 0.0793 | 0.0778 | 0.0556 | 0.0699 | 0.0656 |
| 18. | 0.0732 | 0.0771 | 0.0753 | 0.0739 | 0.0822 | 0.0897 | 0.0771 | 0.0770 |
| 19. | 0.0700 | 0.0637 | 0.0649 | 0.0616 | 0.0684 | 0.0795 | 0.0609 | 0.0571 |
| 20. | 0.0559 | 0.0584 | 0.0544 | 0.0537 | 0.0564 | 0.0705 | 0.0611 | 0.0630 |
| 21. | 0.0709 | 0.0786 | 0.0719 | 0.0751 | 0.0696 | 0.0727 | 0.0749 | 0.0598 |
| 22. | 0.0612 | 0.0648 | 0.0655 | 0.0640 | 0.0647 | 0.0677 | 0.0574 | 0.0682 |
| 23. | 0.0634 | 0.0553 | 0.0552 | 0.0702 | 0.0547 | 0.0514 | 0.0641 | 0.0575 |
| 24. | 0.0658 | 0.0655 | 0.0680 | 0.0625 | 0.0685 | 0.0613 | 0.0594 | 0.0642 |
| 25. | 0.0822 | 0.0906 | 0.0881 | 0.0696 | 0.0844 | 0.0807 | 0.0718 | 0.0772 |
| 26. | 0.0623 | 0.0645 | 0.0615 | 0.0625 | 0.0593 | 0.0609 | 0.0606 | 0.0646 |
| 27. | 0.0534 | 0.0583 | 0.0526 | 0.0500 | 0.0522 | 0.0463 | 0.0572 | 0.0580 |
| 28. | 0.0463 | 0.0482 | 0.0496 | 0.0448 | 0.0427 | 0.0419 | 0.0464 | 0.0476 |
| 29. | 0.0786 | 0.0503 | 0.0675 | 0.0590 | 0.0636 | 0.0691 | 0.0701 | 0.0600 |
| 30. | 0.0531 | 0.0610 | 0.0732 | 0.0831 | 0.0809 | 0.0614 | 0.0738 | 0.0757 |
| 31. | 0.0533 | 0.0503 | 0.0507 | 0.0459 | 0.0432 | 0.0684 | 0.0562 | 0.0514 |
| 32. | 0.0566 | 0.0472 | 0.0478 | 0.0519 | 0.0474 | 0.0499 | 0.0585 | 0.0455 |
| 33. | 0.0724 | 0.0685 | 0.0623 | 0.0690 | 0.0722 | 0.0746 | 0.0725 | 0.0701 |
| 34. | 0.0664 | 0.0597 | 0.0628 | 0.0626 | 0.0606 | 0.0527 | 0.0622 | 0.0682 |
| 35. | 0.0672 | 0.0705 | 0.0617 | 0.0656 | 0.0709 | 0.0640 | 0.0725 | 0.0609 |
| 36. | 0.0596 | 0.0566 | 0.0494 | 0.0613 | 0.0460 | 0.0627 | 0.0594 | 0.0498 |
| 37. | 0.0604 | 0.0619 | 0.0587 | 0.0556 | 0.0504 | 0.0617 | 0.0614 | 0.0576 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 38. | 0.0495 | 0.0481 | 0.0542 | 0.0509 | 0.0452 | 0.0628 | 0.0661 | 0.0572 |
| 39. | 0.0587 | 0.0868 | 0.0707 | 0.0792 | 0.0868 | 0.0678 | 0.0767 | 0.0767 |
| 40. | 0.0568 | 0.0546 | 0.0436 | 0.0660 | 0.0547 | 0.0571 | 0.0551 | 0.0545 |
| 41. | 0.0658 | 0.0661 | 0.0682 | 0.0739 | 0.0651 | 0.0741 | 0.0715 | 0.0790 |
| 42. | 0.0706 | 0.0743 | 0.0625 | 0.0735 | 0.0654 | 0.0658 | 0.0703 | 0.0711 |
| 43. | 0.0802 | 0.0658 | 0.0779 | 0.0800 | 0.0748 | 0.0765 | 0.0782 | 0.0739 |
| 44. | 0.0808 | 0.0743 | 0.0751 | 0.0847 | 0.0778 | 0.0709 | 0.0824 | 0.0816 |
| 45. | 0.0656 | 0.0722 | 0.0733 | 0.0609 | 0.0613 | 0.0578 | 0.0654 | 0.0628 |
| 46. | 0.0728 | 0.0599 | 0.0711 | 0.0669 | 0.0688 | 0.0678 | 0.0689 | 0.0740 |
| 47. | 0.0568 | 0.0505 | 0.0559 | 0.0566 | 0.0627 | 0.0603 | 0.0550 | 0.0605 |
| 48. | 0.0839 | 0.0648 | 0.0756 | 0.0757 | 0.0717 | 0.0763 | 0.0729 | 0.0695 |
| 49. | 0.0709 | 0.0739 | 0.0720 | 0.0736 | 0.0709 | 0.0698 | 0.0675 | 0.0706 |
| 50. | 0.0801 | 0.0755 | 0.0648 | 0.0716 | 0.0678 | 0.0719 | 0.0593 | 0.0608 |
| 51. | 0.0899 | 0.0811 | 0.0885 | 0.0875 | 0.1101 | 0.0825 | 0.0799 | 0.0765 |
| 52. | 0.0385 | 0.0410 | 0.0356 | 0.0372 | 0.0301 | 0.0334 | 0.0321 | 0.0333 |
| 53. | 0.0755 | 0.0819 | 0.0794 | 0.0721 | 0.0912 | 0.0913 | 0.0846 | 0.0818 |
| 54. | 0.0735 | 0.0588 | 0.0587 | 0.0597 | 0.0486 | 0.0660 | 0.0593 | 0.0501 |
| 55. | 0.0693 | 0.0568 | 0.0681 | 0.0601 | 0.0487 | 0.0541 | 0.0580 | 0.0569 |
| 56. | 0.0559 | 0.0517 | 0.0505 | 0.0534 | 0.0598 | 0.0580 | 0.0677 | 0.0518 |
| 57. | 0.0661 | 0.0601 | 0.0567 | 0.0524 | 0.0573 | 0.0514 | 0.0557 | 0.0566 |
| 58. | 0.0709 | 0.0739 | 0.0720 | 0.0736 | 0.0709 | 0.0698 | 0.0675 | 0.0706 |
| 59. | 0.0669 | 0.0648 | 0.0662 | 0.0766 | 0.0670 | 0.0674 | 0.0636 | 0.0650 |
| 60. | 0.0652 | 0.0628 | 0.0619 | 0.0736 | 0.0732 | 0.0730 | 0.0768 | 0.0769 |
| 61. | 0.0598 | 0.0552 | 0.0623 | 0.0638 | 0.0666 | 0.0703 | 0.0672 | 0.0587 |
| 62. | 0.0442 | 0.0383 | 0.0379 | 0.0364 | 0.0484 | 0.0492 | 0.0416 | 0.0411 |
| 63. | 0.0636 | 0.0631 | 0.0532 | 0.0552 | 0.0600 | 0.0609 | 0.0592 | 0.0621 |
| 64. | 0.0785 | 0.0814 | 0.0695 | 0.0727 | 0.0767 | 0.0788 | 0.0723 | 0.0782 |
| 65. | 0.0483 | 0.0568 | 0.0558 | 0.0538 | 0.0546 | 0.0521 | 0.0597 | 0.0515 |
| 66. | 0.0603 | 0.0604 | 0.0706 | 0.0775 | 0.0738 | 0.0650 | 0.0595 | 0.0679 |
| 67. | 0.0624 | 0.0615 | 0.0738 | 0.0764 | 0.0619 | 0.0819 | 0.0707 | 0.0621 |
| 68. | 0.0858 | 0.0726 | 0.0666 | 0.0757 | 0.0671 | 0.0607 | 0.0661 | 0.0703 |
| 69. | 0.0737 | 0.0701 | 0.0688 | 0.0770 | 0.0647 | 0.0748 | 0.0772 | 0.0777 |
| 70. | 0.0672 | 0.0673 | 0.0593 | 0.0683 | 0.0677 | 0.0626 | 0.0622 | 0.0713 |
| 71. | 0.0884 | 0.0865 | 0.0796 | 0.0771 | 0.0847 | 0.0753 | 0.0775 | 0.0798 |

Table 6.3. The values of FSIM for proposed method

| Finger# / Person # | **1** | **2** | **3** | **4** | ***5*** | ***6*** | ***7*** | ***8*** |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.8330 | 0.8332 | 0.8318 | 0.8706 | 0.8551 | 0.8380 | 0.8272 | 0.8403 |
| 2. | 0.8482 | 0.8543 | 0.8566 | 0.8510 | 0.8575 | 0.8522 | 0.8560 | 0.8526 |
| 3. | 0.8805 | 0.8678 | 0.8940 | 0.8827 | 0.8719 | 0.8748 | 0.8880 | 0.8718 |
| 4. | 0.8488 | 0.8467 | 0.8474 | 0.8419 | 0.8435 | 0.8592 | 0.8512 | 0.8703 |
| 5. | 0.8488 | 0.8544 | 0.8487 | 0.8508 | 0.8480 | 0.8495 | 0.8640 | 0.8531 |
| 6. | 0.8533 | 0.8573 | 0.8571 | 0.8591 | 0.8547 | 0.8602 | 0.8627 | 0.8547 |
| 7. | 0.8745 | 0.8599 | 0.8719 | 0.8781 | 0.8663 | 0.8639 | 0.8690 | 0.8733 |
| 8. | 0.8140 | 0.8143 | 0.8239 | 0.8211 | 0.8134 | 0.8278 | 0.8277 | 0.8168 |
| 9. | 0.8640 | 0.8588 | 0.8596 | 0.8736 | 0.8621 | 0.8660 | 0.8660 | 0.8673 |
| 10. | 0.8571 | 0.8606 | 0.8811 | 0.8658 | 0.8617 | 0.8736 | 0.8658 | 0.8599 |
| 11. | 0.8284 | 0.8376 | 0.8496 | 0.8513 | 0.8416 | 0.8577 | 0.8589 | 0.8437 |
| 12. | 0.8690 | 0.8674 | 0.8459 | 0.8436 | 0.8547 | 0.8408 | 0.8529 | 0.8525 |
| 13. | 0.8612 | 0.8608 | 0.8605 | 0.8438 | 0.8438 | 0.8532 | 0.8531 | 0.8574 |
| 14. | 0.8817 | 0.8816 | 0.8779 | 0.8791 | 0.8838 | 0.8816 | 0.8819 | 0.8771 |
| 15. | 0.8721 | 0.8641 | 0.8571 | 0.8646 | 0.8670 | 0.8606 | 0.8755 | 0.8652 |
| 16. | 0.8412 | 0.8283 | 0.8465 | 0.8548 | 0.8368 | 0.8452 | 0.8692 | 0.8611 |
| 17. | 0.8578 | 0.8752 | 0.8720 | 0.8683 | 0.8678 | 0.8796 | 0.8731 | 0.8585 |
| 18. | 0.8563 | 0.8740 | 0.8693 | 0.8708 | 0.8667 | 0.8688 | 0.8779 | 0.8797 |
| 19. | 0.8967 | 0.8939 | 0.8864 | 0.8991 | 0.9003 | 0.9078 | 0.8934 | 0.9024 |
| 20. | 0.8765 | 0.8632 | 0.8579 | 0.8763 | 0.8737 | 0.8922 | 0.8809 | 0.8850 |
| 21. | 0.8638 | 0.8644 | 0.8619 | 0.8537 | 0.8598 | 0.8596 | 0.8550 | 0.8598 |
| 22. | 0.8946 | 0.8969 | 0.9040 | 0.8894 | 0.8882 | 0.8766 | 0.8851 | 0.8793 |
| 23. | 0.8846 | 0.8756 | 0.8811 | 0.8763 | 0.8885 | 0.8826 | 0.8771 | 0.8824 |
| 24. | 0.8644 | 0.8612 | 0.8745 | 0.8625 | 0.8703 | 0.8784 | 0.8829 | 0.8708 |
| 25. | 0.8429 | 0.8421 | 0.8400 | 0.8685 | 0.8364 | 0.8412 | 0.8437 | 0.8455 |
| 26. | 0.8752 | 0.8796 | 0.8911 | 0.8935 | 0.8700 | 0.8677 | 0.8757 | 0.8738 |
| 27. | 0.8722 | 0.8549 | 0.8637 | 0.8648 | 0.8584 | 0.8668 | 0.8633 | 0.8580 |
| 28. | 0.8497 | 0.8574 | 0.8537 | 0.8437 | 0.8470 | 0.8445 | 0.8504 | 0.8454 |
| 29. | 0.8553 | 0.8742 | 0.8671 | 0.8578 | 0.8541 | 0.8537 | 0.8462 | 0.8580 |
| 30. | 0.8728 | 0.8686 | 0.8499 | 0.8520 | 0.8445 | 0.8670 | 0.8558 | 0.8463 |
| 31. | 0.8361 | 0.8478 | 0.8338 | 0.8544 | 0.8444 | 0.8451 | 0.8380 | 0.8385 |
| 32. | 0.8462 | 0.8702 | 0.8561 | 0.8832 | 0.8726 | 0.8647 | 0.8565 | 0.8626 |
| 33. | 0.8537 | 0.8753 | 0.8730 | 0.8532 | 0.8551 | 0.8585 | 0.8606 | 0.8540 |
| 34. | 0.8502 | 0.8535 | 0.8614 | 0.8560 | 0.8560 | 0.8571 | 0.8473 | 0.8519 |
| 35. | 0.8537 | 0.8470 | 0.8349 | 0.8483 | 0.8484 | 0.8361 | 0.8498 | 0.8605 |
| 36. | 0.8379 | 0.8409 | 0.8455 | 0.8377 | 0.8468 | 0.8357 | 0.8339 | 0.8394 |
| 37. | 0.8679 | 0.8754 | 0.8721 | 0.8681 | 0.8528 | 0.8525 | 0.8711 | 0.8508 |
| 38. | 0.8617 | 0.8492 | 0.8623 | 0.8429 | 0.8708 | 0.8399 | 0.8513 | 0.8429 |
| 39. | 0.8853 | 0.8754 | 0.8760 | 0.8525 | 0.8511 | 0.8658 | 0.8533 | 0.8457 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 40. | 0.8847 | 0.8559 | 0.8728 | 0.8719 | 0.8471 | 0.8550 | 0.8543 | 0.8555 |
| 41. | 0.8702 | 0.8696 | 0.8678 | 0.8689 | 0.8646 | 0.8710 | 0.8741 | 0.8743 |
| 42. | 0.8943 | 0.8714 | 0.8698 | 0.8824 | 0.8832 | 0.8781 | 0.8738 | 0.8690 |
| 43. | 0.8735 | 0.8748 | 0.8691 | 0.8563 | 0.8607 | 0.8613 | 0.8613 | 0.8534 |
| 44. | 0.8529 | 0.8714 | 0.8544 | 0.8583 | 0.8553 | 0.8685 | 0.8637 | 0.8553 |
| 45. | 0.8980 | 0.8608 | 0.8610 | 0.8649 | 0.8694 | 0.8621 | 0.8597 | 0.8531 |
| 46. | 0.8620 | 0.8733 | 0.8690 | 0.8664 | 0.8565 | 0.8609 | 0.8566 | 0.8493 |
| 47. | 0.8503 | 0.8532 | 0.8467 | 0.8437 | 0.8423 | 0.8472 | 0.8516 | 0.8450 |
| 48. | 0.8682 | 0.8931 | 0.8813 | 0.8730 | 0.8723 | 0.8658 | 0.8824 | 0.8929 |
| 49. | 0.8716 | 0.8712 | 0.8757 | 0.8742 | 0.8702 | 0.8722 | 0.8692 | 0.8781 |
| 50. | 0.8707 | 0.8681 | 0.8942 | 0.8794 | 0.8839 | 0.8720 | 0.8851 | 0.8877 |
| 51. | 0.8801 | 0.8790 | 0.8852 | 0.8813 | 0.8771 | 0.8811 | 0.8815 | 0.8827 |
| 52. | 0.8666 | 0.8531 | 0.8757 | 0.8696 | 0.8475 | 0.8578 | 0.8589 | 0.8623 |
| 53. | 0.8849 | 0.8695 | 0.8818 | 0.8819 | 0.8507 | 0.8544 | 0.8635 | 0.8692 |
| 54. | 0.8278 | 0.8389 | 0.8386 | 0.8365 | 0.8431 | 0.8357 | 0.8288 | 0.8484 |
| 55. | 0.8810 | 0.8715 | 0.8744 | 0.8798 | 0.8803 | 0.8741 | 0.8706 | 0.8799 |
| 56. | 0.8798 | 0.8970 | 0.8879 | 0.8876 | 0.8864 | 0.8926 | 0.8836 | 0.8883 |
| 57. | 0.8250 | 0.8515 | 0.8439 | 0.8659 | 0.8389 | 0.8365 | 0.8409 | 0.8460 |
| 58. | 0.8716 | 0.8712 | 0.8757 | 0.8742 | 0.8702 | 0.8722 | 0.8692 | 0.8781 |
| 59. | 0.8631 | 0.8666 | 0.8631 | 0.8603 | 0.8637 | 0.8607 | 0.8642 | 0.8722 |
| 60. | 0.8535 | 0.8506 | 0.8596 | 0.8411 | 0.8432 | 0.8503 | 0.8504 | 0.8388 |
| 61. | 0.8630 | 0.8715 | 0.8742 | 0.8614 | 0.8743 | 0.8554 | 0.8612 | 0.8655 |
| 62. | 0.8767 | 0.8525 | 0.8393 | 0.8611 | 0.8739 | 0.8750 | 0.8660 | 0.8485 |
| 63. | 0.8705 | 0.8769 | 0.8891 | 0.8809 | 0.8692 | 0.8806 | 0.8744 | 0.8778 |
| 64. | 0.8758 | 0.8850 | 0.8816 | 0.8797 | 0.8858 | 0.8810 | 0.8822 | 0.8778 |
| 65. | 0.8390 | 0.8372 | 0.8360 | 0.8359 | 0.8255 | 0.8398 | 0.8270 | 0.8377 |
| 66. | 0.8872 | 0.8978 | 0.8851 | 0.8899 | 0.8865 | 0.9022 | 0.8989 | 0.8845 |
| 67. | 0.8739 | 0.8784 | 0.8716 | 0.8676 | 0.8783 | 0.8795 | 0.8757 | 0.8776 |
| 68. | 0.8867 | 0.8823 | 0.8771 | 0.8650 | 0.8746 | 0.8822 | 0.8717 | 0.8878 |
| 69. | 0.8551 | 0.8579 | 0.8689 | 0.8607 | 0.8467 | 0.8482 | 0.8645 | 0.8567 |
| 70. | 0.8702 | 0.8705 | 0.8754 | 0.8643 | 0.8741 | 0.8643 | 0.8802 | 0.8769 |
| 71. | 0.8748 | 0.8755 | 0.8848 | 0.8798 | 0.8841 | 0.8901 | 0.8836 | 0.8803 |

Figure 6.6 shows the Comparison between Eun-Kyung Yun et al. [6] method and the proposed method. Eun-Kyung Yun et al. method is an adaptive pre-processing method which is based on binary image morphology. The proposed method is based on fuzzy morphology, where their image has more information than binary image morphology.

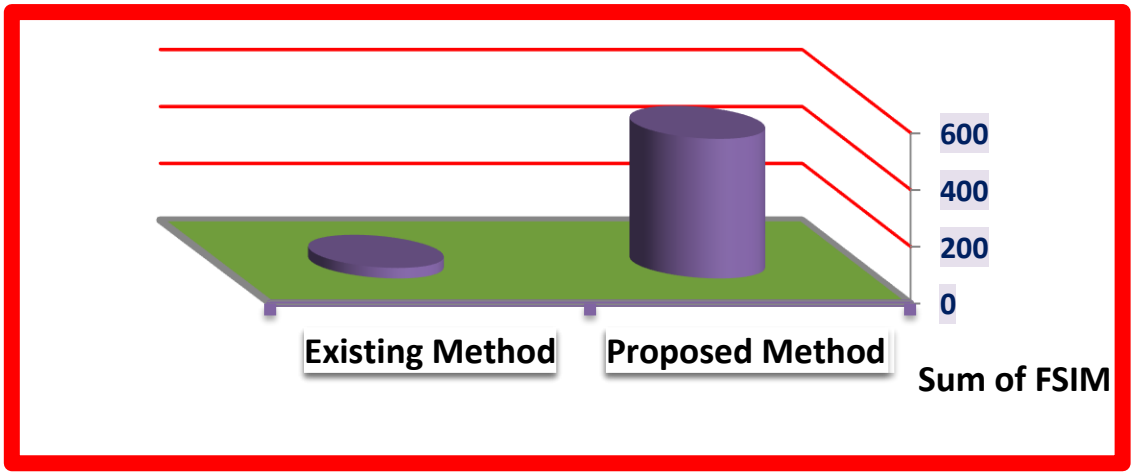


Figure 6.6. Comparison of existing method and proposed method.

From table 6.2, table 6.3, and figure 6.6, it is clear that the proposed method is better than the Eun-Kyung Yun et al. method. The percentage improvement between the proposed method and Eun-Kyung Yun et al. method is 900% (calculated only for table 6.2, and table 6.3). Therefore, the value of FSIM for the proposed method as in table 6.3 is higher than the value of FSIM for the Eun-Kyung Yun et al. method as in table 6.2.

## 6.2. Theorem

Our proposed fuzzy system gives a much higher feature similarity than binary since the proposed system uses fuzzy logic which returns more information about fingerprint images than binary methods.

The proof is as seen in the simulation in figure 6.6. The simulation shows that the proposed method is better than existing method by900%percentage,which is calculated based on table 6.2 and table 6.3.

## 6.3. Summary

In this chapter enhancement of dry fingerprint image was done by smoothing the input image using Winner filter, then fuzzy dilation operation was performed. Fuzzifying the greyscale image was done with the use of the S-fuzzy membership functions. Then a fuzzy structuring element is traversed on the whole image to process dilation operations. The union operation between the original image and dilated image was then performed to come up with an enhanced image. Finally the FSIM was applied to evaluate the performance of the proposed method. The new proposed method did perform 900% better than the existing Binary method.

# CHAPTER VII

## Oily Fingerprint Image Enhancement

Fingerprint identification is the most widely used biometric technology and is used in criminal investigations, commercial applications, etc. With such a wide variety of uses for the technology, the demographics and environment conditions that it is used in are just as diverse. However, the identification performance of such system is very sensitive to the quality of the captured fingerprint image. Fingerprint image quality analysis and enhancement are useful in improving the performance of fingerprint identification systems. In many systems it is preferable to substitute low quality images for better ones. Therefore, image enhancement takes an important part in image processing. Various factors can affect the quality of fingerprint images such as dryness/wetness conditions, non-uniform and inconsistent contact, permanent cuts and so on. Many of these factors cannot be avoided. Therefore, enhancement of the captured fingerprint image is necessary and meaningful. In this chapter we explain the implementation of the oily fingerprint image enhancement by using fuzzy morphology, and lastly measure the performance of the method by using FSIM.

### 7.1 The Method

The method used to enhance an oily fingerprint image is made of five steps as shown in figure 7.1. Figure 7.1 presents the flowchart for the new proposed fingerprint enhancement method. The steps are as follows:

1. Input an image to a low pass filter to smooth the image. Then we employ a fuzzy morphology dilation on the smoothed image and lastly we perform the intersection operation between the original image and dilated image.
2. Employ fuzzy morphology erosion on the original image.
3. Come up with the inverse of the dilated image.
4. Take the intersection operation between the output of second step and third step.
5. Perform the union operation between the output of step number one and number four to display results.

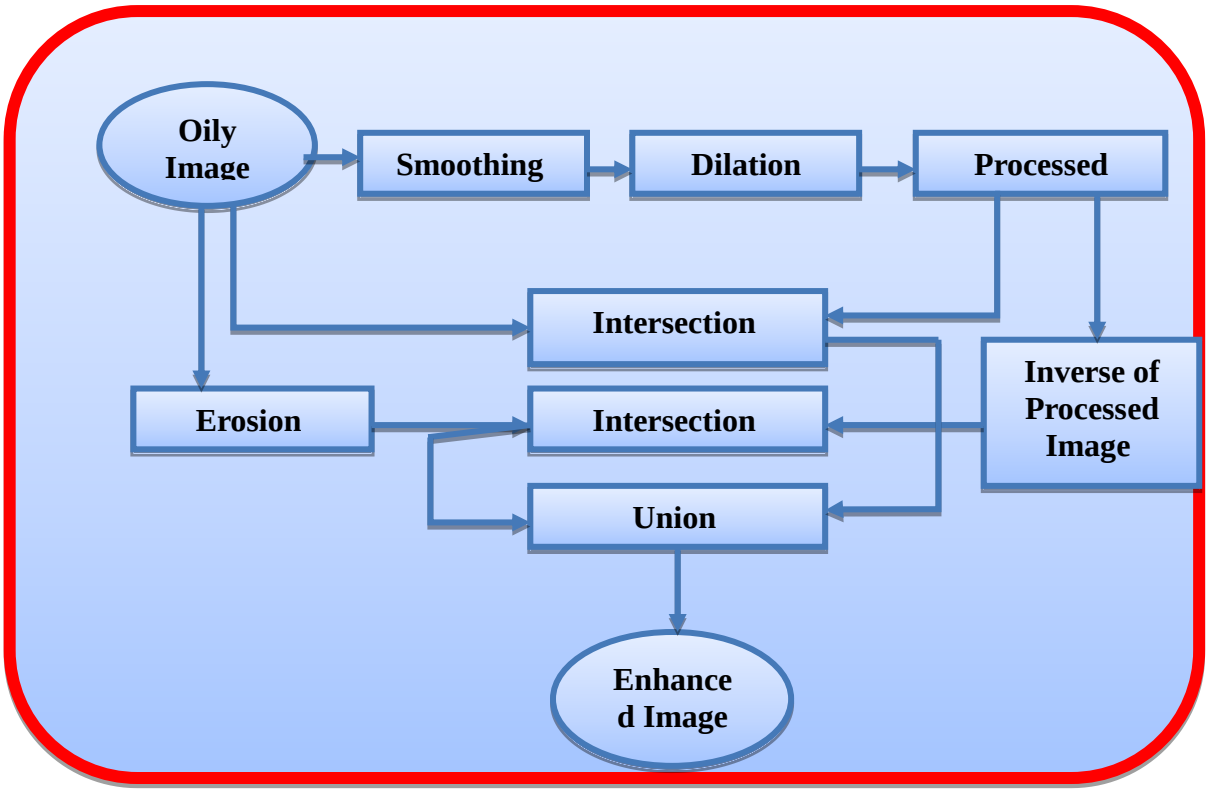


Figure 7.1. The flowchart of the proposed method.

Performance evolution is conducted using feature similarity index metric (FSIM) to measure the quality of the enhanced image compared with previous studies.

In the first step, smoothing is applied to reduce noise in an image and to decrease the disparity between pixel values by using Winner filter. Figure 7.2showsa smoothed fingerprint image.

The next step is to apply fuzzy dilation on the smoothed image, where fuzzy dilation and erosion operations are dual operations. The logical intersection operation is usedto extracts the intersection of black pixels in the inputted oily image and dilated image, as shown in figure 7.1. Figure7.3 shows a fuzzy dilated image.

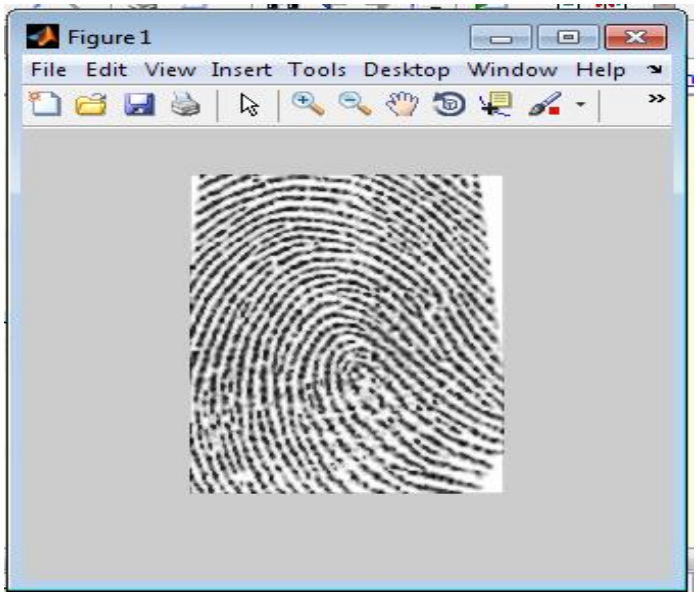


Figure 7.2. A smoothed image.

The next step is to employ fuzzy morphology erosion on the original image. Dilation is applied on smoothing fingerprint image, where erosion is applied on oily image (input image). In the two operations we must fuzzify the input images, then apply both the α-cut dilation equation to dilate the images and the α-cut erosion equation to erode the images. Then defuzzification is applied to transfer the image back to grayscale image.

For image fuzzification, we are required to employ a membership function where the values lie between 0 and 1. The S-membership function is selected, where the *S*-function is the most prominent heuristic membership function. The shape of *S*-function depends on three parameters *a, b,* and *c.* The parameters *a, b* and *c* are specified to ensure the membership function maximizes the information contained in the image. In this study we will applythe*S*-membership function where the parameters a, b, and c are defined as follows:

$$\begin{aligned} \mu(x) &= 0 && x<a, \\ &= 2\,[(x-a)/(c-a)]^2 && a < x < b, \\ &= 1 - 2\,[(x-c)/(c-a)]^2 && b < x < c, \\ &= 1 && x > c, \end{aligned}$$

where, *b* is any value between *a* and *c*. Figure 7.3 shows the membership function plot for *a* = *Xmin*, *c* = *Xmax*, for normalized set.

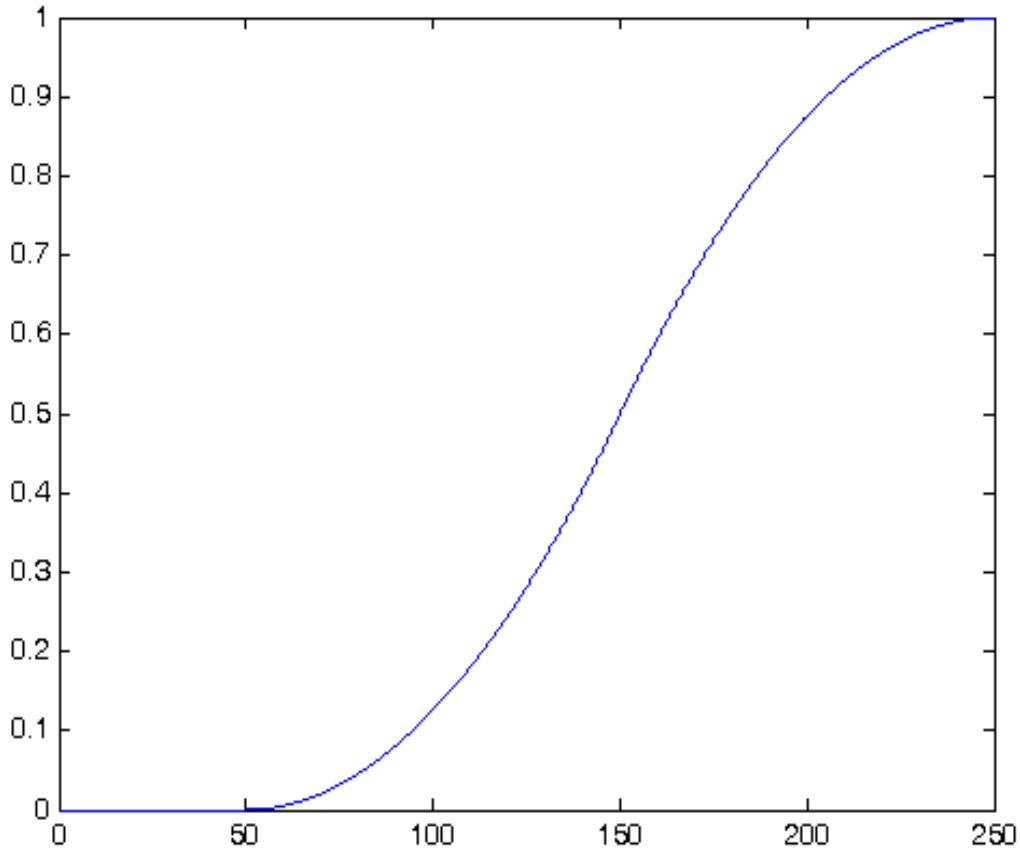


Figure 7.3. S-membership functions.

The fuzzified image can be created from the *S*-membership function with improved quality. The membership function $\mu = 1$ represents the maximum brightness. This study used the*S*-membership function to improve the quality of image.

Fuzzy dilation is applied by probe the structuring element in table 7.1to scan the whole image, and replace the center pixel in each probe of structuring element with pixel that satisfy the following formula:

$$[g(x) \oplus \mu(x)] \quad \alpha(x) = supmin[g(x-y), \mu(x)]$$

where y∈X for Dilation.

$$[g(x) \ominus \mu(x)] \quad \alpha(x) = infmax[1 - g(x-y), \mu(x)]$$

where y∈X for Erosion.

Table7.1. Structuring Element Mask

| | | |
|---|---|---|
| **0.3** | **0.8** | **0.3** |
| **0.8** | **1** | **0.8** |
| **0.3** | **0.8** | **0.3** |

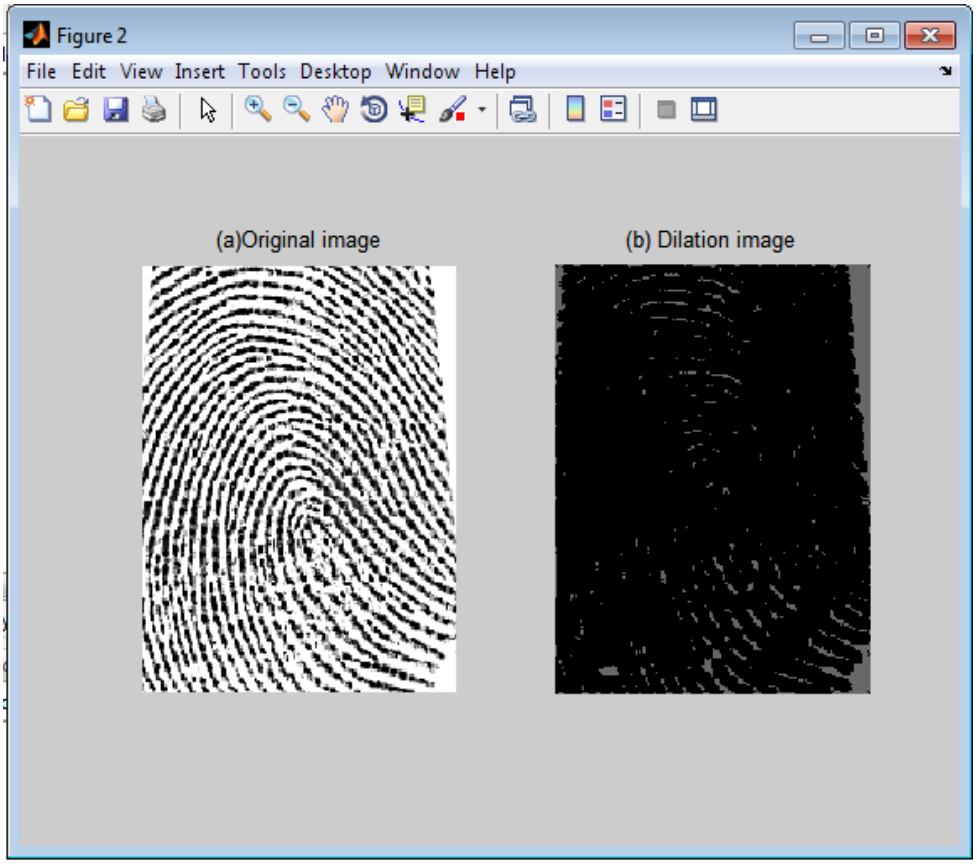


Figure 7.4. Fuzzy dilation image.

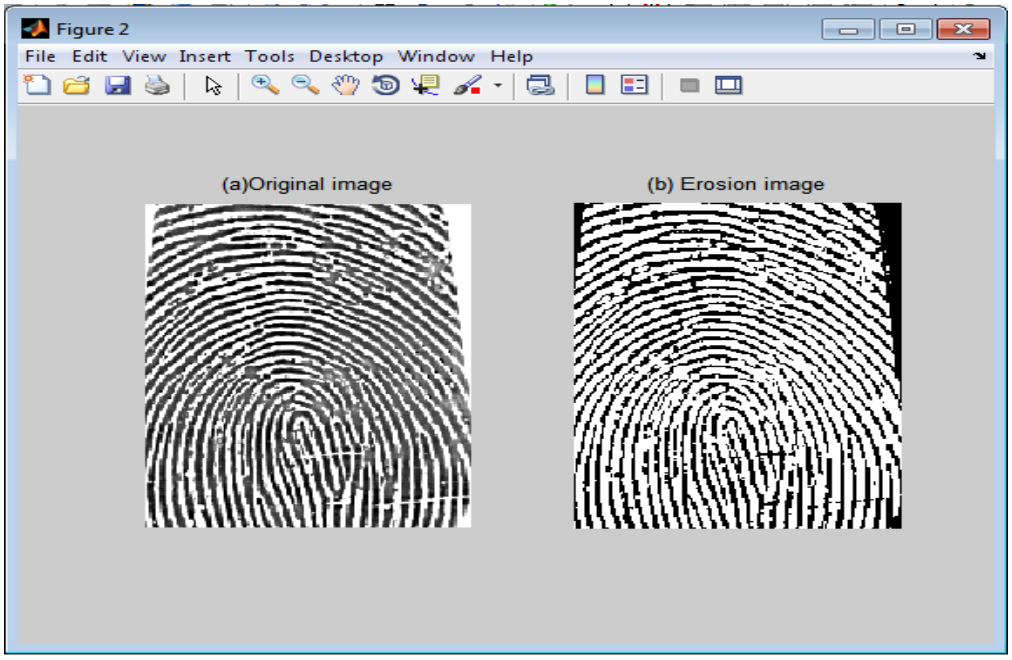


Figure 7.5.Fuzzy erosion fingerprint image.

In the next step the inverse of the dilated image is derived. The Inverse of image is obtained from dilated image using Matlab function which is the subtraction operation between the maximum value in the image and the image pixels as shown in figure 7.6.

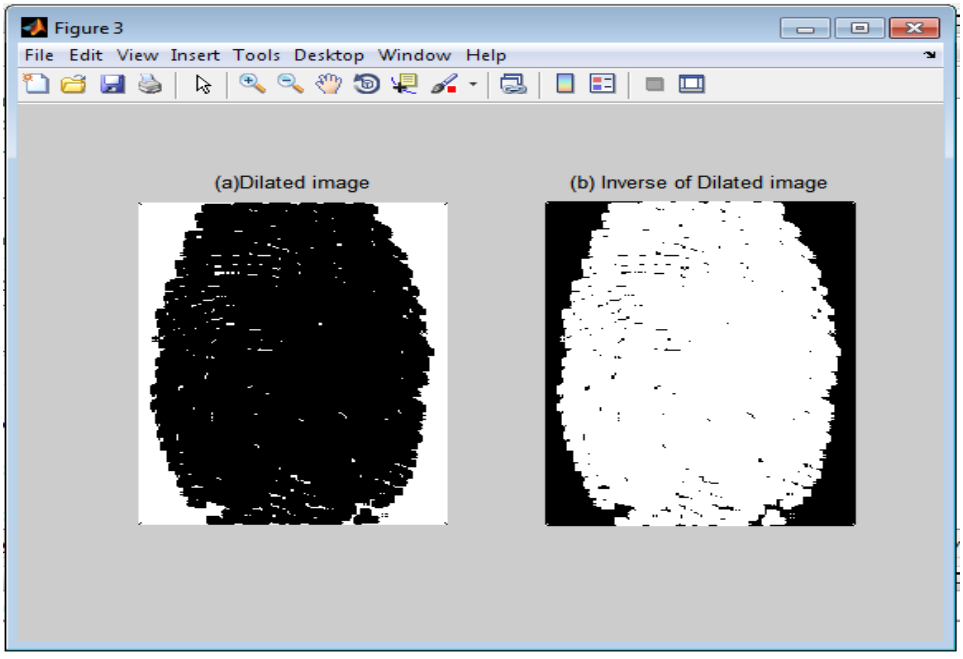


Figure 7.6. The inverse of the dilated image.

In the following step the logical operation intersection and union has been applied. The logical intersection and union operations are used to extract the intersection and union of black pixels, as shown in figure 7.1 (the design of the new proposed fingerprint enhanced method). Figure 7.7 shows the intersection between the erosion image and the inverse of the dilated image.

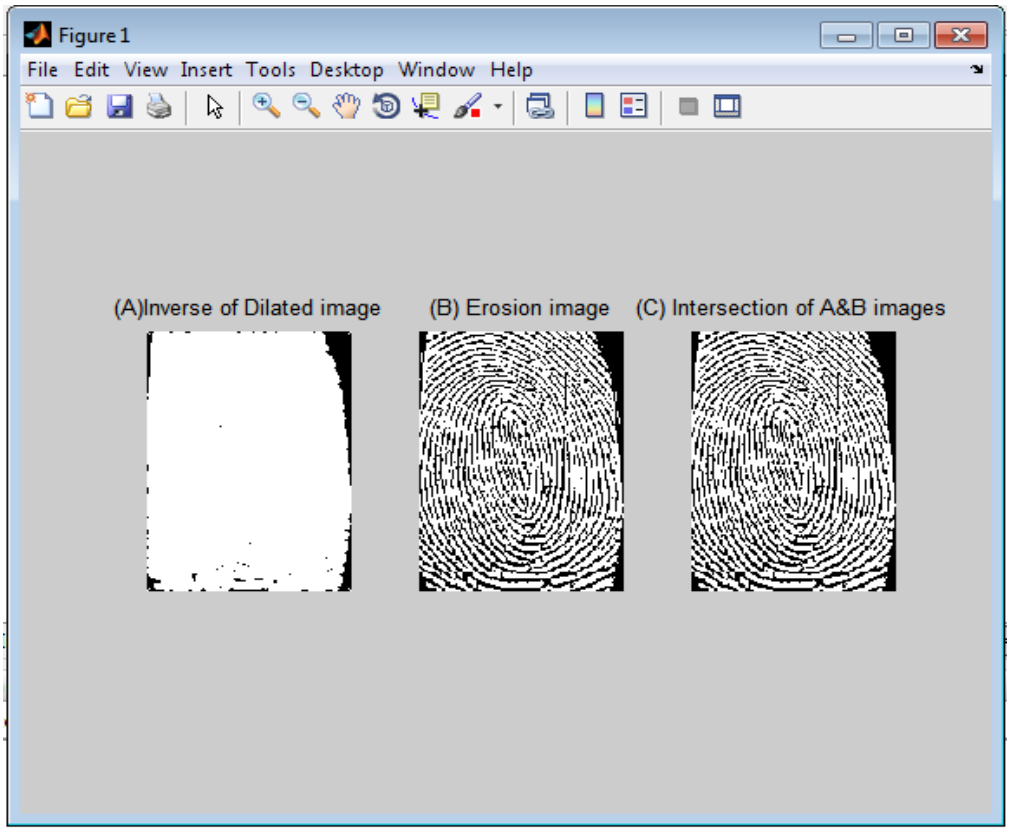


Figure 7.7. Intersection operation.

Finally, we perform the union operation between the two intersection operations presented in the flowchart of the design of the new proposed fingerprint enhanced method. By the end of the operation the valleys are enhanced as shown in figure 7.8.

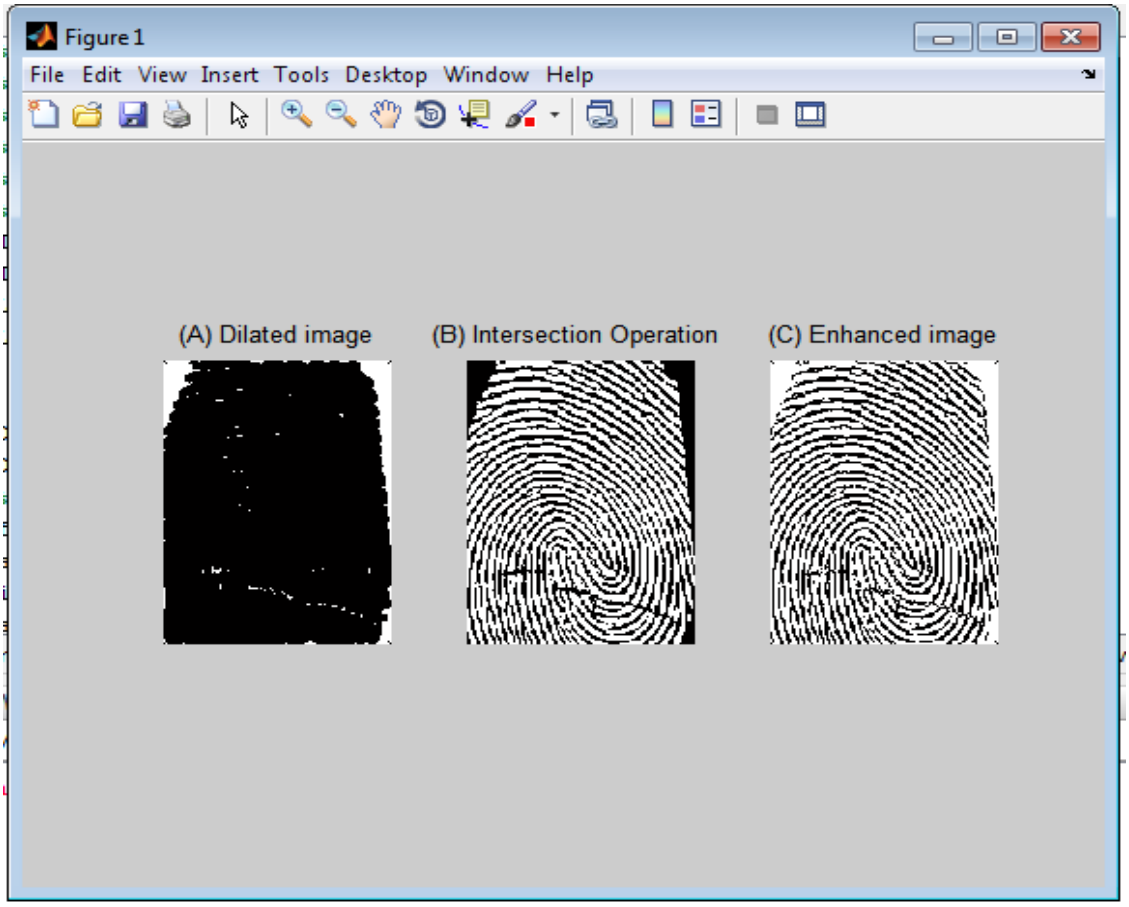


Figure 7.8. Enhanced image.

The proposed oily fingerprint enhancement method using fuzzy morphology gives high FSIM when compared to the existing enhancement method by Eun-Kyung Yun et al. [6](see appendix C)that uses adaptive pre-processing method based on binary morphology image processing. The FSIM for the existing method and the proposed method are described in table 7.2 and table 7.3, respectively.

Table7.2. The FSIM values for Eun-Kyung Yun et al. Method [6]

| Finger# / Person # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.5616 | 0.5807 | 0.5616 | 0.5829 | 0.5928 | 0.5967 | 0.5872 | 0.5793 |
| 2. | 0.6084 | 0.6082 | 0.6122 | 0.5884 | 0.5925 | 0.5924 | 0.5756 | 0.5825 |
| 3. | 0.6013 | 0.6030 | 0.5938 | 0.5997 | 0.5925 | 0.6033 | 0.5915 | 0.6014 |
| 4. | 0.5900 | 0.6087 | 0.5857 | 0.5990 | 0.6244 | 0.6060 | 0.5964 | 0.5998 |
| 5. | 0.5538 | 0.5442 | 0.5723 | 0.5713 | 0.5361 | 0.5759 | 0.5674 | 0.5706 |
| 6. | 0.5928 | 0.5755 | 0.5792 | 0.5942 | 0.5827 | 0.5716 | 0.5646 | 0.5684 |
| 7. | 0.5983 | 0.6157 | 0.6174 | 0.6167 | 0.6164 | 0.6166 | 0.6222 | 0.6233 |
| 8. | 0.6002 | 0.5917 | 0.6112 | 0.6033 | 0.5707 | 0.6020 | 0.6121 | 0.5855 |
| 9. | 0.6247 | 0.6163 | 0.6151 | 0.6163 | 0.6190 | 0.6085 | 0.6238 | 0.6231 |
| 10. | 0.6024 | 0.6161 | 0.6303 | 0.6291 | 0.6112 | 0.6234 | 0.6244 | 0.6316 |
| 11. | 0.6035 | 0.5904 | 0.6008 | 0.5939 | 0.6059 | 0.5908 | 0.5726 | 0.5665 |
| 12. | 0.5878 | 0.5875 | 0.5806 | 0.5857 | 0.5881 | 0.5887 | 0.5927 | 0.5882 |
| 13. | 0.6191 | 0.6084 | 0.6064 | 0.6134 | 0.6168 | 0.6176 | 0.6157 | 0.6227 |
| 14. | 0.5442 | 0.5477 | 0.5465 | 0.5352 | 0.5523 | 0.5580 | 0.5514 | 0.5466 |
| 15. | 0.5684 | 0.5610 | 0.5629 | 0.5760 | 0.5667 | 0.5732 | 0.6007 | 0.5960 |
| 16. | 0.5986 | 0.5891 | 0.5959 | 0.5923 | 0.5856 | 0.5877 | 0.5827 | 0.5938 |
| 17. | 0.5872 | 0.5632 | 0.5776 | 0.5876 | 0.5798 | 0.6004 | 0.5877 | 0.5905 |
| 18. | 0.5800 | 0.5985 | 0.6130 | 0.6094 | 0.6275 | 0.5957 | 0.5952 | 0.5873 |
| 19. | 0.5714 | 0.5710 | 0.5801 | 0.5891 | 0.5806 | 0.5816 | 0.5586 | 0.5843 |
| 20. | 0.5470 | 0.5575 | 0.5599 | 0.5876 | 0.5768 | 0.5550 | 0.5559 | 0.5563 |
| 21. | 0.5801 | 0.5750 | 0.5707 | 0.5786 | 0.5941 | 0.5966 | 0.6043 | 0.5890 |
| 22. | 0.5787 | 0.5937 | 0.5771 | 0.5710 | 0.5732 | 0.5856 | 0.5833 | 0.5748 |
| 23. | 0.5410 | 0.5489 | 0.5535 | 0.5506 | 0.5522 | 0.5492 | 0.5578 | 0.5712 |
| 24. | 0.5890 | 0.6062 | 0.5930 | 0.5951 | 0.5952 | 0.6057 | 0.5876 | 0.5932 |
| 25. | 0.5616 | 0.5723 | 0.5780 | 0.5873 | 0.5680 | 0.5679 | 0.5782 | 0.5547 |
| 26. | 0.6053 | 0.6058 | 0.6104 | 0.5965 | 0.6041 | 0.6057 | 0.6007 | 0.5840 |
| 27. | 0.6176 | 0.6184 | 0.6189 | 0.6026 | 0.6118 | 0.6270 | 0.6239 | 0.6279 |
| 28. | 0.5987 | 0.6040 | 0.5736 | 0.6087 | 0.6097 | 0.6099 | 0.6120 | 0.6042 |
| 29. | 0.5664 | 0.5939 | 0.6079 | 0.6045 | 0.5900 | 0.6114 | 0.6065 | 0.6006 |
| 30. | 0.5906 | 0.5916 | 0.5937 | 0.5872 | 0.5917 | 0.5991 | 0.5844 | 0.5911 |
| 31. | 0.6015 | 0.5918 | 0.6082 | 0.6112 | 0.6032 | 0.6005 | 0.6011 | 0.6056 |
| 32. | 0.5909 | 0.5612 | 0.5736 | 0.5691 | 0.5789 | 0.5683 | 0.5827 | 0.5935 |
| 33. | 0.5652 | 0.5423 | 0.5488 | 0.5531 | 0.5736 | 0.5968 | 0.5973 | 0.5870 |
| 34. | 0.5992 | 0.5980 | 0.6037 | 0.5967 | 0.6023 | 0.6002 | 0.6002 | 0.6032 |
| 35. | 0.5859 | 0.5876 | 0.5952 | 0.5985 | 0.5992 | 0.5956 | 0.6098 | 0.5998 |
| 36. | 0.5912 | 0.6018 | 0.5907 | 0.6028 | 0.6108 | 0.6095 | 0.6055 | 0.6053 |
| 37. | 0.5884 | 0.5758 | 0.5624 | 0.5656 | 0.5700 | 0.5920 | 0.5823 | 0.5991 |
| 38. | 0.5994 | 0.5998 | 0.5979 | 0.5875 | 0.5815 | 0.6184 | 0.6149 | 0.6149 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 39. | 0.5885 | 0.6007 | 0.5848 | 0.5890 | 0.5970 | 0.5884 | 0.5927 | 0.6046 |
| 40. | 0.5442 | 0.5402 | 0.5589 | 0.5514 | 0.5812 | 0.5188 | 0.5580 | 0.5623 |
| 41. | 0.6126 | 0.6234 | 0.6195 | 0.6167 | 0.6116 | 0.5981 | 0.5917 | 0.5952 |
| 42. | 0.5948 | 0.5791 | 0.5723 | 0.5559 | 0.5941 | 0.5973 | 0.6008 | 0.5981 |
| 43. | 0.5601 | 0.6009 | 0.5942 | 0.5685 | 0.5985 | 0.5635 | 0.5902 | 0.5819 |
| 44. | 0.5925 | 0.5982 | 0.5891 | 0.6014 | 0.5980 | 0.5886 | 0.5905 | 0.5933 |
| 45. | 0.6049 | 0.5966 | 0.6005 | 0.6027 | 0.6024 | 0.6039 | 0.6034 | 0.6043 |
| 46. | 0.5150 | 0.5259 | 0.5297 | 0.5372 | 0.5543 | 0.5408 | 0.5458 | 0.5372 |
| 47. | 0.5797 | 0.6120 | 0.5678 | 0.5794 | 0.5726 | 0.5967 | 0.6064 | 0.6044 |
| 48. | 0.5891 | 0.5972 | 0.5886 | 0.5793 | 0.5793 | 0.5760 | 0.5803 | 0.5772 |
| 49. | 0.5833 | 0.5871 | 0.6024 | 0.6044 | 0.5932 | 0.6160 | 0.5954 | 0.5937 |
| 50. | 0.5797 | 0.5824 | 0.5910 | 0.5887 | 0.5881 | 0.5840 | 0.5910 | 0.5810 |
| 51. | 0.5801 | 0.5983 | 0.5890 | 0.6010 | 0.5651 | 0.5695 | 0.6028 | 0.5883 |
| 52. | 0.6068 | 0.6162 | 0.6004 | 0.5910 | 0.6063 | 0.6106 | 0.6118 | 0.6053 |
| 53. | 0.5892 | 0.5795 | 0.5676 | 0.5845 | 0.5803 | 0.5780 | 0.5870 | 0.5803 |
| 54. | 0.5982 | 0.5948 | 0.5917 | 0.6139 | 0.5796 | 0.5878 | 0.5853 | 0.5888 |
| 55. | 0.5911 | 0.5987 | 0.5939 | 0.5941 | 0.5901 | 0.5863 | 0.5931 | 0.5989 |
| 56. | 0.5965 | 0.5853 | 0.5886 | 0.6013 | 0.6017 | 0.6043 | 0.6055 | 0.5760 |
| 57. | 0.5958 | 0.5991 | 0.6021 | 0.6056 | 0.6007 | 0.5978 | 0.5960 | 0.6007 |
| 58. | 0.5833 | 0.5871 | 0.6024 | 0.6044 | 0.5932 | 0.6160 | 0.5954 | 0.5937 |
| 59. | 0.6183 | 0.6012 | 0.5818 | 0.5846 | 0.6024 | 0.5948 | 0.6082 | 0.5882 |
| 60. | 0.5787 | 0.5515 | 0.5485 | 0.5443 | 0.5760 | 0.5640 | 0.5706 | 0.5564 |
| 61. | 0.5997 | 0.6008 | 0.6065 | 0.6024 | 0.6048 | 0.6102 | 0.6197 | 0.6068 |
| 62. | 0.6026 | 0.6014 | 0.5965 | 0.6074 | 0.6151 | 0.6145 | 0.6077 | 0.5934 |
| 63. | 0.5719 | 0.6131 | 0.6148 | 0.6127 | 0.6231 | 0.6127 | 0.6136 | 0.6107 |
| 64. | 0.5942 | 0.5881 | 0.5877 | 0.5911 | 0.5812 | 0.5591 | 0.5885 | 0.5844 |
| 65. | 0.5829 | 0.5908 | 0.5772 | 0.5768 | 0.5826 | 0.5917 | 0.5927 | 0.5865 |
| 66. | 0.5883 | 0.5773 | 0.5805 | 0.5766 | 0.5846 | 0.5751 | 0.5858 | 0.5686 |
| 67. | 0.5825 | 0.6176 | 0.6029 | 0.6086 | 0.5992 | 0.5654 | 0.5866 | 0.6075 |
| 68. | 0.5662 | 0.5614 | 0.5623 | 0.6009 | 0.6216 | 0.5936 | 0.6048 | 0.6128 |
| 69. | 0.5705 | 0.6054 | 0.5977 | 0.6093 | 0.6028 | 0.5918 | 0.5940 | 0.5998 |
| 70. | 0.5936 | 0.5913 | 0.5812 | 0.5979 | 0.5909 | 0.5963 | 0.5939 | 0.5851 |
| 71. | 0.5823 | 0.5922 | 0.5802 | 0.5729 | 0.5815 | 0.6089 | 0.5901 | 0.5782 |

Table7.3. The FSIM values for the proposed method.

| Finger# / Person # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.7291 | 0.6977 | 0.7291 | 0.7038 | 0.6884 | 0.7126 | 0.7226 | 0.6786 |
| 2. | 0.7201 | 0.7192 | 0.7137 | 0.6996 | 0.6969 | 0.6977 | 0.6916 | 0.6880 |
| 3. | 0.7032 | 0.7084 | 0.7042 | 0.7091 | 0.7157 | 0.7092 | 0.7078 | 0.7207 |
| 4. | 0.7356 | 0.7179 | 0.7466 | 0.7393 | 0.7339 | 0.7344 | 0.7384 | 0.7239 |
| 5. | 0.6561 | 0.6531 | 0.6599 | 0.6716 | 0.6498 | 0.6836 | 0.6661 | 0.6770 |
| 6. | 0.6983 | 0.6689 | 0.6769 | 0.6784 | 0.6902 | 0.6859 | 0.6452 | 0.6470 |
| 7. | 0.7064 | 0.6978 | 0.7300 | 0.7203 | 0.7101 | 0.7239 | 0.7261 | 0.7175 |
| 8. | 0.7551 | 0.7550 | 0.7484 | 0.7443 | 0.7273 | 0.7198 | 0.7288 | 0.7197 |
| 9. | 0.7323 | 0.7086 | 0.7234 | 0.7232 | 0.7179 | 0.7298 | 0.7161 | 0.7369 |
| 10. | 0.7181 | 0.7314 | 0.7348 | 0.7343 | 0.7251 | 0.7163 | 0.7386 | 0.7352 |
| 11. | 0.7224 | 0.7087 | 0.7011 | 0.6993 | 0.7033 | 0.6781 | 0.6669 | 0.6829 |
| 12. | 0.7173 | 0.7129 | 0.7222 | 0.7221 | 0.7241 | 0.7246 | 0.7135 | 0.7190 |
| 13. | 0.7325 | 0.7497 | 0.7368 | 0.7435 | 0.7459 | 0.7384 | 0.7571 | 0.7368 |
| 14. | 0.6684 | 0.6898 | 0.6778 | 0.6806 | 0.6876 | 0.6904 | 0.6813 | 0.6856 |
| 15. | 0.6473 | 0.6726 | 0.6626 | 0.6763 | 0.6718 | 0.6744 | 0.6906 | 0.7029 |
| 16. | 0.7455 | 0.7348 | 0.7380 | 0.7562 | 0.7493 | 0.7489 | 0.7291 | 0.7689 |
| 17. | 0.7188 | 0.6813 | 0.6952 | 0.6989 | 0.7213 | 0.7342 | 0.7302 | 0.7340 |
| 18. | 0.6763 | 0.6522 | 0.6957 | 0.7039 | 0.7060 | 0.6929 | 0.6856 | 0.6619 |
| 19. | 0.6589 | 0.6677 | 0.6659 | 0.6712 | 0.6831 | 0.6846 | 0.6688 | 0.6772 |
| 20. | 0.7228 | 0.7454 | 0.7358 | 0.7236 | 0.7345 | 0.7459 | 0.7401 | 0.7264 |
| 21. | 0.6482 | 0.6686 | 0.6415 | 0.6706 | 0.6844 | 0.6806 | 0.6787 | 0.6874 |
| 22. | 0.6682 | 0.6736 | 0.6710 | 0.6807 | 0.6706 | 0.6858 | 0.6756 | 0.6694 |
| 23. | 0.6539 | 0.6585 | 0.6647 | 0.6724 | 0.6538 | 0.6623 | 0.6819 | 0.6812 |
| 24. | 0.7204 | 0.7084 | 0.6897 | 0.7023 | 0.7261 | 0.7308 | 0.7124 | 0.7170 |
| 25. | 0.6393 | 0.6584 | 0.6450 | 0.6525 | 0.6489 | 0.6530 | 0.6545 | 0.6474 |
| 26. | 0.7193 | 0.7342 | 0.7129 | 0.7207 | 0.7224 | 0.7182 | 0.7194 | 0.7080 |
| 27. | 0.7203 | 0.7251 | 0.7239 | 0.7146 | 0.7190 | 0.7349 | 0.7278 | 0.7277 |
| 28. | 0.7676 | 0.7563 | 0.7194 | 0.7493 | 0.7418 | 0.7408 | 0.7579 | 0.7365 |
| 29. | 0.6504 | 0.6835 | 0.6945 | 0.6948 | 0.6849 | 0.6933 | 0.7116 | 0.6910 |
| 30. | 0.6647 | 0.6601 | 0.6656 | 0.6666 | 0.6624 | 0.6738 | 0.6491 | 0.6808 |
| 31. | 0.6944 | 0.6961 | 0.7277 | 0.7197 | 0.7164 | 0.7170 | 0.7119 | 0.7170 |
| 32. | 0.7028 | 0.6843 | 0.6909 | 0.6787 | 0.6882 | 0.6788 | 0.7018 | 0.7155 |
| 33. | 0.6723 | 0.6418 | 0.6656 | 0.6674 | 0.6751 | 0.6869 | 0.6933 | 0.7059 |
| 34. | 0.7286 | 0.7149 | 0.7142 | 0.7172 | 0.7330 | 0.7394 | 0.7389 | 0.7316 |
| 35. | 0.7087 | 0.6990 | 0.7002 | 0.7140 | 0.7228 | 0.7116 | 0.7203 | 0.7358 |
| 36. | 0.7118 | 0.7126 | 0.7213 | 0.7268 | 0.7239 | 0.7335 | 0.7343 | 0.7325 |
| 37. | 0.6913 | 0.6933 | 0.6964 | 0.6899 | 0.6984 | 0.6984 | 0.7123 | 0.7192 |
| 38. | 0.7157 | 0.7314 | 0.7059 | 0.7060 | 0.7077 | 0.7186 | 0.7419 | 0.7328 |
| 39. | 0.7107 | 0.7201 | 0.7097 | 0.7331 | 0.7177 | 0.7275 | 0.7289 | 0.7337 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 40. | 0.6667 | 0.6659 | 0.6755 | 0.6859 | 0.6966 | 0.6319 | 0.6746 | 0.6832 |
| 41. | 0.6860 | 0.6901 | 0.6953 | 0.6827 | 0.6841 | 0.6626 | 0.6482 | 0.6729 |
| 42. | 0.6445 | 0.6650 | 0.6313 | 0.6549 | 0.6641 | 0.6693 | 0.6736 | 0.6494 |
| 43. | 0.6663 | 0.6996 | 0.6993 | 0.6709 | 0.6971 | 0.6800 | 0.6917 | 0.6881 |
| 44. | 0.6771 | 0.6754 | 0.6703 | 0.6825 | 0.7004 | 0.6914 | 0.6976 | 0.6899 |
| 45. | 0.7021 | 0.6933 | 0.7070 | 0.7010 | 0.7109 | 0.7168 | 0.7027 | 0.7006 |
| 46. | 0.6308 | 0.6367 | 0.6389 | 0.6381 | 0.6565 | 0.6597 | 0.6503 | 0.6502 |
| 47. | 0.7046 | 0.7319 | 0.7113 | 0.7180 | 0.7201 | 0.7386 | 0.7511 | 0.7440 |
| 48. | 0.6942 | 0.7140 | 0.6991 | 0.6674 | 0.6918 | 0.6851 | 0.6868 | 0.7103 |
| 49. | 0.6563 | 0.6637 | 0.6955 | 0.6844 | 0.6791 | 0.6965 | 0.6686 | 0.6658 |
| 50. | 0.6880 | 0.7060 | 0.7076 | 0.6977 | 0.7252 | 0.6992 | 0.7024 | 0.6892 |
| 51. | 0.6641 | 0.6700 | 0.6738 | 0.6781 | 0.6481 | 0.6590 | 0.6810 | 0.6596 |
| 52. | 0.7317 | 0.7257 | 0.7232 | 0.7046 | 0.7157 | 0.7230 | 0.7119 | 0.7046 |
| 53. | 0.6625 | 0.6530 | 0.6364 | 0.6745 | 0.6606 | 0.6508 | 0.6697 | 0.6698 |
| 54. | 0.7473 | 0.7271 | 0.7370 | 0.7545 | 0.7346 | 0.7305 | 0.7155 | 0.7161 |
| 55. | 0.7188 | 0.7153 | 0.7067 | 0.7183 | 0.7024 | 0.7098 | 0.7105 | 0.7128 |
| 56. | 0.7019 | 0.7111 | 0.7055 | 0.7081 | 0.7104 | 0.7159 | 0.6832 | 0.6679 |
| 57. | 0.7120 | 0.7516 | 0.7396 | 0.7175 | 0.7566 | 0.7135 | 0.7211 | 0.7566 |
| 58. | 0.6563 | 0.6637 | 0.6955 | 0.6844 | 0.6791 | 0.6965 | 0.6686 | 0.6658 |
| 59. | 0.6813 | 0.7031 | 0.6858 | 0.6806 | 0.6933 | 0.6843 | 0.7004 | 0.6948 |
| 60. | 0.6724 | 0.6643 | 0.6659 | 0.6867 | 0.6965 | 0.6616 | 0.6919 | 0.6961 |
| 61. | 0.7220 | 0.7094 | 0.7179 | 0.7367 | 0.7164 | 0.7086 | 0.7390 | 0.7325 |
| 62. | 0.6616 | 0.7306 | 0.7395 | 0.7084 | 0.7207 | 0.7496 | 0.7151 | 0.7549 |
| 63. | 0.6519 | 0.7200 | 0.7142 | 0.6939 | 0.7195 | 0.7048 | 0.6997 | 0.7005 |
| 64. | 0.6608 | 0.6641 | 0.6703 | 0.6638 | 0.6284 | 0.6674 | 0.6394 | 0.6655 |
| 65. | 0.7197 | 0.7090 | 0.7192 | 0.7270 | 0.7347 | 0.7117 | 0.7338 | 0.7274 |
| 66. | 0.6771 | 0.6743 | 0.6847 | 0.6779 | 0.6917 | 0.6819 | 0.6939 | 0.6721 |
| 67. | 0.6920 | 0.7237 | 0.7176 | 0.7307 | 0.7140 | 0.6858 | 0.7085 | 0.7396 |
| 68. | 0.6504 | 0.6410 | 0.6490 | 0.7166 | 0.7089 | 0.7082 | 0.7028 | 0.7127 |
| 69. | 0.6613 | 0.6911 | 0.6664 | 0.6965 | 0.6654 | 0.6914 | 0.7113 | 0.7134 |
| 70. | 0.7127 | 0.6748 | 0.6839 | 0.6976 | 0.6969 | 0.7130 | 0.6997 | 0.6837 |
| 71. | 0.6727 | 0.7160 | 0.6995 | 0.6974 | 0.6883 | 0.6686 | 0.7193 | 0.7132 |

Figure 7.8 shows the comparison between Eun-Kyung Yun et al. [6] method and the new proposed method. Eun-Kyung Yun et al. method is an adaptive pre-processing method which is based on binary image morphology. The proposed method is based on fuzzy morphology, where the image has more information than the binary image morphology.

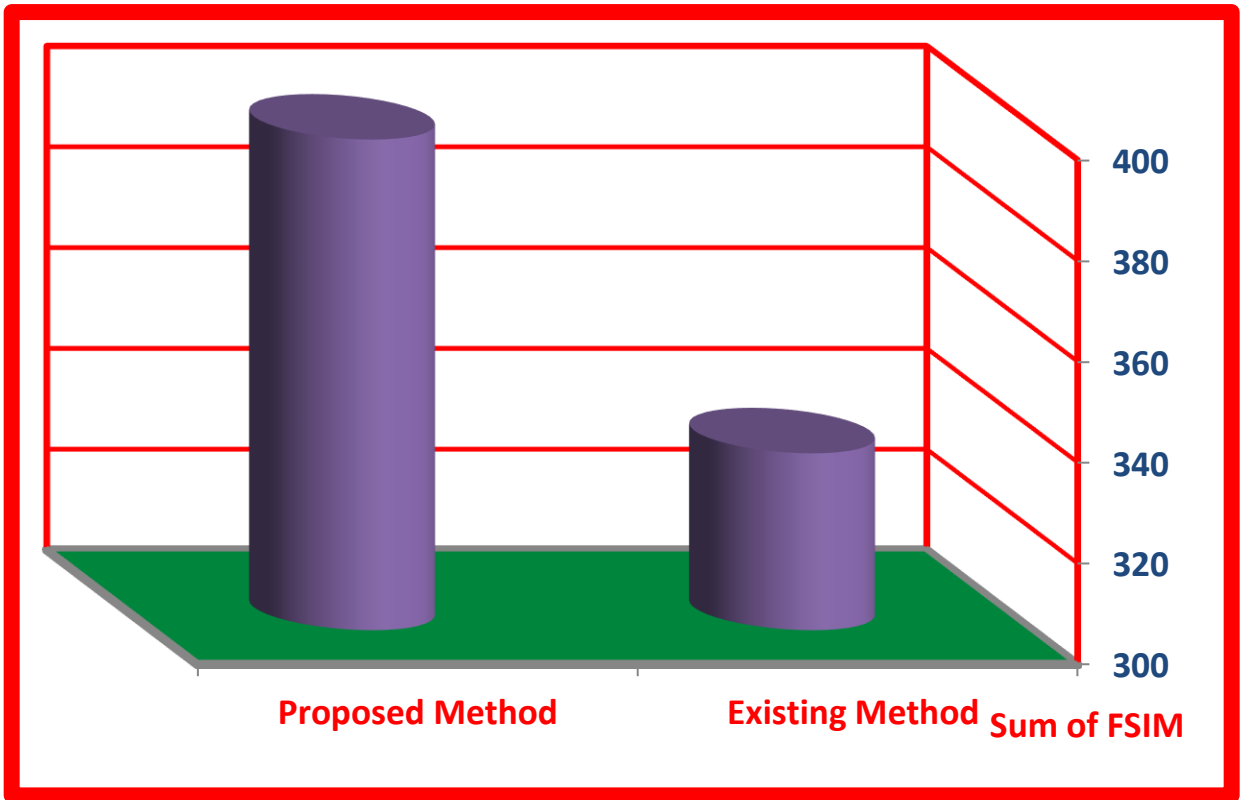


Figure 7. 9. Comparison of existing method and proposed method.

From table 7.2, table 7.3, and figure.7.9, it is clear that the proposed method is performing better than the Eun-Kyung Yun et al. method. The percentage improvement between the proposed method and Eun-Kyung Yun et al. method is 118.75% (calculated only for table 7.2, and table 7.3). Therefore, the value of FSIM for the proposed method as in table7.3 is higher than the value of FSIM for the Eun-Kyung Yun et al. method as in table 7.2.

7.2**Theorem**

The proposed fuzzy system gives a high feature similarity than binary since the proposed system using fuzzy logic which returns more information about fingerprint images than binary methods.

The proof is as seen from the simulation shown in figure 7.9. The simulation shows that the proposed method is better than the existing method by a percentage of 118.75%, which is calculated from table 7.2 and table 7.3.

.

## 7.3**Summary**

In this chapter enhancing the oily fingerprint image by smoothing the image using Winner filter was presented. Fuzzy dilation and erosion operations were performed. The greyscale image was fuzzified with the use of the S-fuzzy membership functions. Then a fuzzy structuring element is traversed on the whole image to process dilation and erosion operations. Then the intersection operation was performed between original image and dilated image.

The inverse of dilated image and dilated image were generated, and the intersection between inversed image and erosion image was done. Finally the union operation between two intersection operations were performed.

# CHAPTER VIII

# Conclusion& Recommendation &Future work

In general, the fingerprint image quality relies on the clearness of separated ridges by valleys and the uniformity of the separation. Although the change in environmental conditions such as temperature, humidity and pressure might influence a fingerprint image in many ways, but the condition of skin still dominate the overall quality of the fingerprint.
Dry skin tends to cause inconsistent contact of the finger ridges with the scanner's platen surface, causing broken ridges and many white pixels replacing ridge structure. To the contrary, the valleys on the oily skin tend to fill up with moisture, causing them to appear black in the image similar to ridge structure.
Therefore, fingerprint image enhancement is an essential preprocessing step in fingerprint recognition applications. The Image enhancement is a preprocessing technique to make the image clearer than the original image, which is needed to increase the contrast between ridges and furrows and for connecting the false broken points of ridges due to insufficient amount of ink or other reasons. High quality fingerprint image is very important for fingerprint verification to work properly. Therefore, the main objective of fingerprint enhancement techniques to process an input image and make it more suitable for identification.

## 8.1. Conclusion

The quality of fingerprint image greatly affects the performance of minutiae extraction and the process of matching in fingerprint identification system. In order to improve the performance of the fingerprint identification system, a fuzzy techniques is used for both quality analysis and enhancement. The dissertation did achieve the following results:
First, the quality analysis performed by extracting four features from a fingerprint image which are the local clarity score (LCS), global clarity score (GCS), ridge_valley thickness ratio (RVTR), and the Global Contrast Factor (GCF). A fuzzy logic technique that uses Mamdani fuzzy rule model was designed. The fuzzy inference system was able to analyze

and determine the fingerprint image type (oily, dry or neutral) based on the extracted feature values and fuzzy inference rules.

The implementation of fuzzy technique was done using MATLAB. A code was written to extract the four features LCS, GLS, RVTR, and GCF and then run this code after saving it as a .m file in MATLAB workspace. Fuzzy if-then rules then was developed and implemented with 63 rules.

The experimental result obtained using the fuzzy technique were successful, and was able to determine the fingerprint image quality (Oily, Neutral, or Dry) according to their input features.

Secondly, fuzzy morphology was applied to enhance dry and oily fingerprint images. For dry fingerprint images, a new algorithm was designed which did include three steps; smoothing, fuzzy dilation, and union operation.

For oily fingerprint images, a new algorithm was designed which also included the following steps;

1. Smooth the input image, then apply fuzzy dilation on the smoothed image and lastly perform the intersection operation between the original image and dilated image.
2. Employ fuzzy erosion on the original image.
3. Come up with the inverse of the dilated image.
4. Take the intersection operation between the output of second step and third step.
5. Perform the union operation between the output of first step and fourthstep to display results.

Fuzzy morphology method improves the quality of fingerprint image, thus, improving the performance of the fingerprint identification system significantly.

All the experimental was done for both quality analysis and image enhancement using the db_its_2009 database which is a private database collected by the department of electrical engineering, institute of technology sepuluh nopember Surabaya.

The performance evaluation was done by using FSIM. Where the Feature Similarity index (FSIM) [16] is an Image Quality Assessment (IQA) metric, which uses computational models to measure the image quality consistently with subjective evaluations.

## 8.2. Recommendation

The research work undertaken in this dissertation has focused on developing a novel technique for analyzing and enhancing dry and oily fingerprint images, as well as reducing the computation time in performing both the enhancement and analysis.

Making the analysis and enhancement real time would enable the following to be done:

- Imbedded fingerprint image quality analysis system inside the fingerprint recognition system without negatively impacting the computation time for whole system.
- Imbedded fingerprint image enhancement system for both dry and oily inside the fingerprint recognition system without negatively impacting the computation time for whole system.

Adaptive fuzzy morphology filter is needed to reduce noise in an image and to decrease the disparity between pixel values, since the fuzzy logic image processing has more accurate information than binary image processing.

## 8.3. Future work

The future research work would be as follows:

- Take the computation time factor in the processes of developing the system of quality analysis and enhancement, which is it enable to
  - ❖ Imbedded fingerprint image quality analysis system inside the fingerprint recognition system.
  - ❖ Imbedded fingerprint image enhancement system for both dry and oily inside the fingerprint recognition system.
- Come up with an adaptive fuzzy morphology filter, to reduce noise in an image and to decrease the disparity between pixel values, since the fuzzy logic image processing has more information than binary image processing.
- To apply the algorithms that were introduced in this dissertation to real world applications such as getting access to an automobile, accessing once ATM bank system, etc.

# References


[1] Jain, Anil K., Arun Ross, and Salil Prabhakar. "An introduction to biometric recognition." Circuits and Systems for Video Technology, IEEE Transactions on14, no. 1 (2004): 4-20.

[2] Zheng, Li, Han Zhi, and Fu Bo. "A Novel Method for the Fingerprint Image Quality Evaluation." In Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on, pp. 1-4. IEEE, 2009.

[3] Selvi, M., and Aloysius George. "FBFET: Fuzzy Based Fingerprint Enhancement Technique based on adaptive thresholding" In Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on, pp. 1-5. IEEE, 2013.

[4] S. N. Sivanandam, S. Sumathi and S. N. Deepa.” Introduction to Fuzzy Logic using MatLab” Springer-Verlag Berlin Heidelberg 2007.

[5] Zheng, L., Zhi, H. and Bo, F., 2009, December. "A Novel Method for the Fingerprint Image Quality Evaluation" In Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on (pp. 1-4). IEEE.

[6] Yun, Eun-Kyung, and Sung-Bae Cho. "Adaptive Fingerprint Image Enhancement with Fingerprint Image Quality Analysis." Image And Vision Computing 24.1 (2006): 101-110.

[7] Sang Keun Oh, Joon Jae Lee, Chul Hyun Park, Bum Soo Kim, and Kil Houm Park. “New Fingerprint Image Enhancement Using Directional Filter Bank” Journal of WSCG, Vol.11, No.1. ISSN 1213-6972, February 3-7, 2003.

[8] SJ, Mrs Preethi, and Mrs K. Rajeswari,"Membership Function modification for Image Enhancement using fuzzy logic", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS). Volume 2, Issue 2, March – April 2013.

[9] Gamassi, Marco, Massimo Lazzaroni, Mauro Misino, Vincenzo Piuri, Daniele Sana, and Fabio Scotti. "Accuracy and performance of biometric systems." In Instrumentation and Measurement Technology Conference, 2004. IMTC 04. Proceedings of the 21st IEEE, vol. 1, pp. 510-515. IEEE, 2004.

[10] D. Kumar and Y. Ryu, “A Brief Introduction of Biometrics and Fingerprint Payment Technology”, International Journal of Advanced Science and Technology, Vol. 4, March, 2009, pp. 25-38.

[11] Institute for the Protection and Security of the Citizen.” Fingerprint Recognition for Children” JRC Technical Reports.September 2013.

[12] Tarun Mahashwari, and Amit Asthana. "Image Enhancement Using Fuzzy Technique." International Journal of Research Review in Engineering Science & Technology (Issn 2278–6643) Volume-2, Issue-2, June-2013.

[13] Pandey, Saurabh. "Study and implementation of morphology for image segmentation" Diss. THAPAR UNIVERSITY PATIALA, 2010.

[14] Pahsa, Alper. "Morphological image processing with fuzzy logic." Journal of Aeronautics and space Technologies 2.3 (2006): 27-34.

[15] Papasaika-Hanusch, Haris. "Digital image processing using matlab." Institute of Geodesy and Photogrammetry, ETH Zurich 63 (1967).

[16] Zhang, L., Zhang, L., Mou, X. and Zhang, D., 2011. FSIM: a feature similarity index for image quality assessment. IEEE transactions on Image Processing, 20(8), pp.2378-2386.

[17] Purneet Kaur, and Jaspreet Kaur, “A Review Paper on Fingerprint Image Enhancement with Different Methods”, International Journal of Modern Engineering Research, Vol. 3, Issue. 4, Jul – Aug. 2013.

[18] Sarrafzadeh, Abdolhossein, Fatemeh Rezazadeh, and Jamshid Shanbehzadeh, "Brightness Preserving Fuzzy Dynamic Histogram Equalization", proceedings of the International Multi-Conference of Engineers and Computer Scientists, vol. 1. 2013.

[19] Magudeeswaran, V., and C. G. Ravichandran. "Fuzzy logic-based histogram equalization for image contrast enhancement." Mathematical Problems in Engineering 2013 (2013).

[20] Selia, I. Golda, And Latha Parthiban,"Approaches For Enhancing Fingerprint Images Using Filters: A Case Study", ACS-International Journal In Computational Intelligence, Vol-2, Issue-1 Apr 2011.

[21] Hong, Lin, Yifei Wan, and Anil Jain. "Fingerprint image enhancement: algorithm and performance evaluation." Pattern Analysis and Machine Intelligence, IEEE Transactions on 20, no. 8 (1998): 777-789.

[22] Balaji, S., and N. Venkatram. "Filtering Of Noise In Fingerprint Images."International Journal of System and Technologies (IJST) 1, No. 1: 87-94. 2008.

[23] Sreedhar, K., and B. Panlal. "Enhancement of Images Using Morphological Transformation." International Journal of Computer Science & Information Technology (IJCSIT) Vol, 4, NO 1, Feb (2012).

[24] Sepasian, M., W. Balachandran, and C. Mares. "Image enhancement for fingerprint minutiae-based algorithms using CLAHE, standard deviation analysis and sliding neighborhood." In Proceedings of the World congress on Engineering and Computer Science, pp. 22-24. 2008.

[25] Longkumer, Nungsanginla, Mukesh Kumar, A. K. Jaiswal, and Rohini Saxena. "Contrast Enhancement Using Various Statistical Operations and Neighborhood Processing." Signal (2014).

[26] Raju Sonavane, Dr.B.S. Sawant. "Noisy fingerprint image enhancement technique for image analysis: A structure similarity measure approach."International Journal of computer science and Network Security (IJCSNS) 7, no. 9 September (2007): 225.

[27] Khalefa, Mustafa Salah, Zaid Amin Abduljabar, and Huda Ameer Zeki. "Fingerprint Image Enhancement By Develop Mehtre Technique." Advanced Computing: An International Journal (ACIJ), 2 (6), 171-182. (2011).

[28] Lu, Haiping, Xudong Jiang, and Wei-Yun Yau. "Effective and efficient fingerprint image post-processing." In Control, Automation, Robotics and Vision, 2002. ICARCV 2002. 7th International Conference on, vol. 2, pp. 985-989. IEEE, 2002.

[29] Kumar, U. Pavan, and P. Padmaja "Image Enhancement Using Adaptive Filtering." International Journal of Engineering Trends and Technology (IJETT)- Volume 6 Number 1 Dec 2013.

[30] Chaurasia, Om Preeti. "An approach to Fingerprint image pre-processing."International Journal of Image, Graphics and Signal Processing (IJIGSP) 4, no. 6 (2012): 29.

[31] Naresh Kumar, and Parag Verma." Fingerprint Image Enhancement And MinutiaMatching" International Journal of Engineering Sciences & Emerging Technologies, June 2012.

[32] Hanoon, Muna F. "Contrast Fingerprint Enhancement Based on Histogram Equalization Followed By Bit Reduction of Vector Quantization." International Journal of Computer Science and Network Security 11, no. 5 (2011): 116-123.

[33] Kanpariya Nilam, and Rahul Joshi. "Adaptive Fingerprint Image Enhancement Techniques and Performance Evaluations" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3 Issue 1, January 2014.

[34] Sang Keun Oh, Joon Jae Lee, Chul Hyun Park, Bum Soo Kim, and Kil Houm Park. "New Fingerprint Image Enhancement Using Directional Filter Bank" Journal of WSCG, Vol.11, No.1. ISSN 1213-6972, February 3-7, 2003.

[35] Deshmukh, Rucha Ashok, and S. A. Ladhake. "A Novel Approach of Using Laplacian Image Enhancement." International Journal Of Advanced Research In Computer Engineering & Technology (IJARCET) Volume 2, Issue 7, July 2013.

[36] Bhowmik, Pankaj, Kishore Bhowmik, Mohammad Nurul Azam, and Mohammed Wahiduzzaman Rony. "Fingerprint Image Enhancement and It " S Feature Extraction for Recognition." International Journal Of Scientific & Technology Research Volume 1, Issue 5, June 2012.

[37] Chikkerur, Sharat, Venu Govindaraju, and Alexander N. Cartwright. "Fingerprint image enhancement using STFT analysis." In Pattern Recognition and Image Analysis, pp. 20-29. Springer Berlin Heidelberg, 2005.

[38] JitendraChoudhary, Dr.Sanjeev Sharma and Jitendra Singh Verma" A New Framework for improving low Quality Fingerprint Images" Int. J. Comp. Tech. Appl., Vol 2 (6),1859-1866. IJCTA | NOV-DEC 2011.

[39] T. Vidhya and T. K. Thivakaran" Fingerprint Image Enhancement using Wavelet over Gabor filters" Int.J.Computer Technology & Applications,Vol 3 (3), 1049-1054, IJCTA| MAY-JUNE 2012.

[40] Patil, Vikas D., and Sachin D. Ruikar. "PCA based image enhancement in wavelet domain." International Journal of Engineering Trends and Technology-Volume3, Issue1-2012 (2012).

[41] Narayana, Battula RVS, and K. Nirmala. "Image Resolution Enhancement by Using Stationary and Discrete Wavelet Decomposition." International Journal of Image, Graphics and Signal Processing (IJIGSP) 4, no. 11 (2012): 41.

[42] Mohammedsayeemuddin, Shaikh, Sima K. Gonsai, and Dharmesh Vandra. "Efficient Fingerprint Image Enhancement Algorithm Based On Gabor Filter." International Journal of Research in Engineering and Technology, Volume 3, Issue 4, Apr 2014.

[43] Babatunde, Iwasokun G., Akinyokun O. Charles, Alese B. Kayode, and Olabode Olatubosun. "A Multi-Level Model for Fingerprint Image Enhancement." In Proceedings of the International Multi-Conference of Engineers and Computer Scientists, vol. 1. 2012.

[44] V.Aarthy, R. Mythili, and M.Mahendran. "Low Quality Fingerprint Image Using Spatial and Frequency Domain Filter" International Journal of Computational Engineering Research. Vol. 2 Issue. 6, October| -2012.

[45] SJ, Mrs Preethi, and Mrs K. Rajeswari. "Membership Function modification for Image Enhancement using fuzzy logic."International Journal of Emerging Trends & Technology in Computer Science (IJETTCS). Volume 2, Issue 2, March – April 2013.

[46] Hasikin, Khairunnisa, and Nor Ashidi Mat Isa. "Enhancement of the low contrast image using fuzzy set theory." In Computer Modelling and Simulation (UKSim), 2012 UKSim 14th International Conference on, pp. 371-376. IEEE, 2012.

[47] Cheng, Heng-Da, and Huijuan Xu. "A Novel Fuzzy Logic Approach to Contrast Enhancement." Pattern Recognition 33, No. 5 (2000): 809-819.

[48] Kundra, Harish, Er Aashima, and Er Monika Verma. "Image Enhancement Based On Fuzzy Logic." International Journal of Computer Science and Network Security 9, No. 10 (2009): 141-145.

[49] Selvi, M., and Aloysius George. "FBFET: Fuzzy Based Fingerprint Enhancement Technique based on adaptive thresholding" In Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on, pp. 1-5. IEEE, 2013.

[50] Diwakar Shrivastava, Vineet Richhariya. "Adaptive Contrast Image Enhancement Based on Fuzzy Set Theory", International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), Volume 4, Issue 2, February 2014.

[51] Suple, Nutan Y., and Sudhir M. Kharad. "Basic approach to image contrast enhancement with fuzzy inference system." International Journal of Scientific and Research Publications 3, no. 6 (2013).

[52] M.Suneel, K.Samba Siva Rao, M.Lavanya and M.Sai Sasanka, "Fuzzy Enhancement Technique Using S-Membership Function In Medical Applications "International Journal of Electronic and Communication Engineering (IJECE) Issue 2278-9901, vol.2,Issue 2,May 2013,121-126.

[53] Hanmandlu, Madasu, and Devendra Jha. "An Optimal Fuzzy System for Color Image Enhancement." Image Processing, IEEE Transactions On 15, No. 10 (2006): 2956-2966.

[54] Popa, Camelia, Mihaela Gordan, Aurel Vlaicu, Bogdan Orza, and Gabriel Oltean. "Computationally efficient algorithm for fuzzy rule-based enhancement on JPEG compressed color images." WSEAS Transactions on Signal Processing 4, no. 5 (2008): 310-319.

[55] Perumal, S. Arumuga, T. C. Rajakumar, and N. Krishnan. "Fuzzy based Filtering Approach on Histogram Specification for Fingerprint Image Enhancement." International Journal of Computer Applications 23, no. 8 (2011).

[56] Choi, YoungSik, and Raghu Krishnapuram. "A robust approach to image enhancement based on fuzzy logic." Image Processing, IEEE Transactions on6, no. 6 (1997): 808-825.

[57] Vlachos, Ioannis K., and George D. Sergiadis. "Parametric indices of fuzziness for automated image enhancement." Fuzzy sets and systems 157, no. 8 (2006): 1126-1138.

[58] Khandelwal, Manglesh, Shweta Saxena, and Priya Bharti. "An efficient algorithm for Image Enhancement." Indian Journal of Computer Science and Engineering (IJCSE) 2 (2005): 118-123.

[59] Alexey Saenko, Galina Polte, and Victor Musalimov, "Image enhancement and image quality analysis using fuzzy logic techniques" Communications (COMM), 2012 9th International Conference 21-23 June 2012. Publisher: IEEE.

[60] Syam, Rahmat, Mochamad Hariadi, and Mauridhi Hery Purnomo. "Determining the Standard Value of Acquisition Distortion of Fingerprint Images Based on Image Quality." ITB Journal of Information and Communication Technology, 4 (2), 115-132. (2010).

[61] Syam, Rahmat, Mochamad Hariadi, and Mauridhi Hery Purnomo. "Determining the Dry Parameter of Fingerprint Image Using Clarity Score and Ridge-valley Thickness Ratio." IAENG International Journal of Computer Science 38, no. 4 (2011): 350-357.

[62] Kukula, Eric, Stephen Elliott, Hakil Kim, and Cristina San Martin. "The impact of fingerprint force on image quality and the detection of minutiae." InElectro/Information Technology, 2007 IEEE International Conference on, pp. 432-437. IEEE, 2007.

[63] Li, Qing-Rong, and Mei Xie. "Measuring Fingerprint Image Quality Using the Fourier Spectrum."Journal of Electronic Science and Technology of China, Vol 5, NO 3, September 2007.

[64] Modi, Shimon K., and Stephen J. Elliott. "Impact of image quality on performance: Comparison of young and elderly fingerprints." In Proceedings of the 6th International Conference on Recent Advances in Soft Computing (RASC 2006), K. Sirlantzis (Ed.), pp. 449-45. 2006.

[65] Matkovic, Kresimir, László Neumann, Attila Neumann, Thomas Psik, and Werner Purgathofer. "Global Contrast Factor-a New Approach to Image Contrast." Computational Aesthetics 2005 (2005): 159-168.

[66] T Samajdar, MI Quraishi. "Analysis and evaluation of image quality metricsInformation, Systems Design and Intelligent Applications" Springer, January 2015

# Appendices

## Appendix A

## The MALAB code For Features Extraction

**Ridge Valley Thickness Ratio**

```
clear all;
h = waitbar(0,'Please Wait a minute...');
%FolderDestination =['D:\PHD_COURCES\Fourth
Sem\DB_ITS_2009\Oily\segmented\'];
Numberof=0;
for x=1 : 71
if x<10
            Front = ['0' num2str(x)];
else
            Front = [num2str(x)];
end
        display('Ratio Thickness for Person Number');disp(x);
for y=1:8
            Numberof=Numberof+1;
            display('Ratio Thickness for finger
Number');disp(y);
if y<10
              Last = ['0' num2str(y)];
else
            Last = [num2str(y)];
end
            FileName=[Front '-N-' Last '.bmp'];
%=============================================================
G=imread(FileName);
G=double(G);
[m n] = size(G);  % m = Total NO of rows in image ; n = Total
NO of coloum in image
w=9;
u=9;
m_blok=m/w; % Total NO of rows in blocks
n_blok=n/u; % Total NO of coloums in blocks
for i=1:m_blok
    G_L = (i-1)*w; G_M = G_L+w;     % G_M = koordinat akhir
baris
    G_L = G_L+1;                    % G_L = koordinat awal
baris
for j=1:n_blok
        G_K = (j-1)*u; G_W = G_K+u;     % G_W = koordinat
akhir kolom
```

```
        G_K = G_K+1;                    % G_K = koordinat
awal kolom
        I_Blok=G(G_L:G_M,G_K:G_W); % Image hasil pembagian
per_blok
        [T_Blok L_Blok]=size(I_Blok);
        RR=0;
for u=1:T_Blok
for v=1:L_Blok
                RR(u,v)=I_Blok(u,v)/(T_Blok*L_Blok);% ratio
for each pixels in image

end
end
        Ratio_perBlok(i,j)=mean2(RR);% ratio per block
        Ratio(i,j)=Ratio_perBlok(i,j)/(m*n);  % thickness
ratio for ridge-valley perblock
end
end
RidgeValley_Ratio=mean2(Ratio);  % caculation of thickness
ratio of ridge-valley
RatioThickness=mean2(RidgeValley_Ratio);
%=============================================================
disp(RatioThickness);
waitbar(Numberof/(71*8));
end
end
close(h)
```

**ClarityScore(LCS GCS)**

```
clear all;
h = waitbar(0,'Please Wait a minute...');
%FolderDestination =['D:\PHD_COURCES\Fourth
Sem\DB_ITS_2009\Oily\segmented\'];
Numberof=0;
for x=1 : 71
if x<10
            Front = ['0' num2str(x)];
else
            Front = [num2str(x)];
end
        display('GCS & LCS for Person Number');disp(x);
for y=1:8
            Numberof=Numberof+1;
            display('GCS & LCS for finger Number');disp(y);
if y<10
              Last = ['0' num2str(y)];
else
```

```
            Last = [num2str(y)];
end
            FileName=[Front '-N-' Last '.bmp'];
%=============================================================

G=imread(FileName);
% G=imread('01-O-01.bmp');
G=double(G);
w=32;
u=13;
[m n] = size(G);  % m = Total NO of rows in image ; n = Total
NO of coloum in image
m_blok=m/w; % Total NO of rows in blocks
n_blok=n/u; % Total NO of coloums in blocks
V3=[];
LCS=[];
GCS_1=[];
for i=1:m_blok
    G_L = (i-1)*w; G_M = G_L+w;      % G_M = koordinat akhir
baris
    G_L = G_L+1;                     % G_L = koordinat awal
baris
for j=1:n_blok
        G_K = (j-1)*u; G_W = G_K+u;      % G_W = koordinat
akhir kolom
        G_K = G_K+1;                     % G_K = koordinat
awal kolom
        V2=G(G_L:G_M,G_K:G_W); % Image hasil pembagian
per_blok
        [T_Blok L_Blok]=size(V2);

for p = 1 : T_Blok
            Tot=0;
for q = 1 : L_Blok
                Tot = Tot + V2(p,q);
end
            V3(p)=Tot/u;
end
%           V4=[V4 V3];
        nil_min=min(V3); nil_max=max(V3);
        DT1= nil_min + (nil_max-nil_min)/2;

        V2=V2';
        Vb=0;
        Rb=0;
for a = 1 : u
for b = 1 : w
```

```
if V2(a,b)<DT1
                    Vb=Vb+1;         % ridge
else
                    Rb=Rb+1;         % valley
end
end
end

        Vt=0;
        Rt=0;
for c = 1 : m
for d = 1 : n
if G(c,d)<DT1
                    Vt=Vt+1;       % Ridge Keseluruhan
else
                    Rt=Rt+1;        % Valley Keseluruhan
end
end
end
        Valley_Score= (Vt);
       Ridge_Score= (Rt);
% Vb adalah banyaknya piksel valley yang rusak yang
berintensitas lebih kecil daripada DT1
% Vt adalah jumlah keseluruhan piksel valley
% Rb adalah banyaknya piksel yang rusak dalam ridge yang
berintensitas lebih tinggi dari DT1
% Rt adalah jumlah keseluruhan piksel ridge
% alpha adalah bagian piksel-piksel yang rusak
%
=================================================================

        alpha=Vb/Vt;
        beta=Rb/Rt;
        LCS=(alpha+beta-0.003)/2 ;
        LCS1=LCS;

        GCS_1(i,j)=LCS;
end
end
GCS=mean2(GCS_1);
%disp(LCS1);
 disp(GCS);
%=================================================================
waitbar(Numberof/(71*8));
end
end
close(h)
```

**GlobalContrastFactor**

```
function [GCF, LC] = getGlobalContrastFactor( im )
% im=imread('01-N-01.bmp');
% GCF = getGlobalContrastFactor( im )
%
% MATLAB algorithm implementation of the
% "Global contrast factor-a new approach to image contrast"
% (Matkovic, Kresimir et al., 2005)
% http://www.cg.tuwien.ac.at/research/publications/2005/matkovic-2005-
glo/
%   Input:
%       im  - image in grayscale
%
%   Output:
%       GCF - global contrast factor
%

% 9 different resolution levels

    GCF = 0.0;

    resolutions = [1 2 4 8 16 25 50 100 200];

    LC = zeros(size(resolutions));

    W = size(im,2);
    H = size(im,1);

    rIm = im;

for i=1:length(resolutions)

%attempt at resizing as in the paper
if i>1
          rIm = imresize(im, 1/(2^(i-1)), 'bilinear');
end

        W = size(rIm,2);
        H = size(rIm,1);

        rL = zeros(size(rIm));
% compute linear luminance l
        l = (double(rIm(:,:))/255) * 2.2;

% compute perceptual luminance L
        rL(:,:) = 100 * sqrt(l);

% compute local contrast for each pixel
        lc = 0.0;
for x=1:H
for y=1:W

if (x == 1) && (x == H)
```

```
if (y == 1) && (y == W)
                        lc = lc + 0;
elseif (y == 1)
                        lc = lc + abs(rL(x, y) - rL(x,y+1));
elseif (y == W)
                         lc = lc + abs(rL(x, y) - rL(x,y-1));
else
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                 abs(rL(x, y) - rL(x,y+1)) )/2;
end

elseif (x == 1)
if (y == 1) && (y == W)
                         lc = lc + abs(rL(x, y) - rL(x+1,y));
elseif (y == 1)
                         lc = lc + ( abs(rL(x, y) - rL(x,y+1)) + ...
                                 abs(rL(x, y) - rL(x+1,y)) )/2;
elseif (y == W)
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                 abs(rL(x, y) - rL(x+1,y)) )/2;
else
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                 abs(rL(x, y) - rL(x,y+1)) + ...
                                 abs(rL(x, y) - rL(x+1,y)) )/3;
end

elseif (x == H)
if (y == 1) && (y == W)
                         lc = lc + abs(rL(x, y) - rL(x-1,y));
elseif (y == 1)
                         lc = lc + ( abs(rL(x, y) - rL(x,y+1)) + ...
                                 abs(rL(x, y) - rL(x-1,y)) )/2;
elseif (y == W)
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                 abs(rL(x, y) - rL(x-1,y)) )/2;
else
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                     abs(rL(x, y) - rL(x,y+1)) + ...
                                     abs(rL(x, y) - rL(x-1,y)) )/3;
end
else% x > 1 && x < H
if (y == 1) && (y == W)
                         lc = lc + ( abs(rL(x, y) - rL(x+1,y)) + ...
                                 abs(rL(x, y) - rL(x-1,y)) )/2;
elseif (y == 1)
                         lc = lc + ( abs(rL(x, y) - rL(x,y+1)) + ...
                                     abs(rL(x, y) - rL(x+1,y)) + ...
                                     abs(rL(x, y) - rL(x-1,y)) )/3;
elseif (y == W)
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                     abs(rL(x, y) - rL(x+1,y)) + ...
                                     abs(rL(x, y) - rL(x-1,y)) )/3;
else
                         lc = lc + ( abs(rL(x, y) - rL(x,y-1)) + ...
                                     abs(rL(x, y) - rL(x,y+1)) + ...
                                     abs(rL(x, y) - rL(x-1,y)) + ...
```

```
                                        abs(rL(x, y) - rL(x+1,y)) )/4;
end

end
end
end

% compute average local contrast c
        c(i) = lc/(W*H);
        w(i) = (-0.406385*(i/9)+0.334573)*(i/9)+ 0.0877526;

% compute global contrast factor
        LC(i) = c(i)*w(i);
        GCF = GCF + LC(i);
end
end
```

## Appendix B

## The MALAB code For Dry Fingerprint Image

```
clear all;close all;
% h=waitbar (0,'please wait 5 minute ....');
% folderdestination = ['D:\DISK1\DB_ITS_2009\DRY\'];
% numberof=0;
% for x=1:71
%     if x<10
%         front=['0' num2str(x)];
%     else
%         front=[num2str(x)];
%     end
%     for y=1:8
%        numberof=numberof+1;
%     if y<10
%         last=['0' num2str(y)];
%     else
%         last=[num2str(y)];
%     end
%     filename=[front '-D-' last '.bmp'];
%****************************************

Imj=imread('03-D-05.bmp');
L=double(Imj);
%****************************************
L=wiener2(Imj,[3  3]);
 figure(1);imshow(L);
%****************************************

I=L;
[H_Image ,W_Image]=size(I);
mid=round(H_Image/2);
NewMin =0;

%% Fuzzification
for i=1 : H_Image
for j=1 : W_Image
if ((I(i,j) >= 0)&& (I(i,j)<= NewMin ))
            I (i,j)=0;
elseif (((I(i,j) >=NewMin)&& (I(i,j)<= mid)))
            I(i,j)=((1/(mid- NewMin)*NewMin) + (1/((mid-NewMin)*
I(i,j))));
end
if (((I(i,j) >= mid) && (I(i,j)<=H_Image)))
            I(i,j)=(1/(((H_Image- mid)*mid) + (1/(H_Image-mid)*I(i,j))));
end
if  ((I(i,j) >=H_Image) && (I(i,j)<= 255 ))
            I (i,j)=1;
end

end
end
```

```
%% you can change image input
% A=imread('01-O-01.bmp');
ImageResult=I;
%I=double(I);
% A=double(A);
[ImageHeight ImageWidth]=size(I);

%% structuring element as Mask Matrix
B=[0.3 0.8 0.3; 0.8 1 0.8; 0.3 0.8 0.3];

%% You can Modify Structuring Element of B
% for example like below
% B=[1 150 125 150 1 ; 125 150 200 150 125; 125 150 200 150 125; 125 150
200 150 125;1 150 125 150 1];

%% it is necessary to identify Mask height and width
[MaskHeight MaskWidth]=size(B);

%% compute start iteration
Start=round(MaskHeight/2);

%% compute the last iteration
Last=fix(MaskHeight/2);

%% image iteration
for j=Start:ImageHeight-Last
for k=Start:ImageWidth-Last
        %% capture image corresponding to the structuring element
        PartOfMatrixA=I(j-Last:j+Last, k-Last:k+Last);
        Index=0;
        %% Masiking Iteration
for m=1:MaskHeight
for n=1:MaskHeight
                Index=Index+1;
                %% compare Matrix A with Structuring element B
                %% And put to DummyMin(Index)
if (PartOfMatrixA(m,n)>B(m,n))
                    DummyMin(Index)=PartOfMatrixA(m,n);
else
                    DummyMin(Index)=B(m,n);
end
end
end

        %% Select max value from the minum values obtained of
DummyMin(Index)
        ImageResult(j,k)= min(DummyMin);
end
end

%% Display the original image and result
% imshow([uint8(A) ImageResult]);
K=ImageResult;
% %**************************************
% %Defuzzification
```

```
for i=1 : H_Image
for j=1 : W_Image
if ((K(i,j) >= 0) &&(K(i,j) <= NewMin))
            J(i,j)= K(i,j);
end
if  ((K(i,j) >= NewMin) &&(K(i,j) <= mid))
            J(i,j)=(mid-NewMin)*(K(i,j)+NewMin);
end
if  ((K(i,j) >= mid) &&(K(i,j) <= H_Image));
            J(i,j)=(H_Image-mid)*(K(i,j)+mid);
end
end
end
figure (2);
subplot(1,2,1);
imshow(Imj);title('(a)Original image');
subplot(1,2,2);
imshow(J);title('(b) Dilation image');

% %*******************************************

D= max(L,J);
figure (3);
subplot(1,2,1);
imshow(Imj);title('(a)Original image');
subplot(1,2,2);
imshow(D);title('(b) Enhanced image');

% %*******************************************

% newmiddle=[front '-DD-' last  '.bmp'];
% newfilename=[folderdestination  newmiddle];
% imwrite(D,newfilename);
% waitbar(numberof/(71*8));
%     end
% end
```

## Appendix C

## The MALAB code For Oily Fingerprint Image

# Step 1:

```
clear all;close all;
h=waitbar (0,'please wait 5 minute ....');
folderdestination = ['D:\DISK1\DB_ITS_2009\OILY\Dilation1\'];
numberof=0;
for x=1:71
if x<10
        front=['0' num2str(x)];
else
        front=[num2str(x)];
end
for y=1:8
       numberof=numberof+1;
if y<10
        last=['0' num2str(y)];
else
        last=[num2str(y)];
end
    filename=[front '-O-' last '.bmp'];
%*****************************************************
Imj=imread(filename);
%L=double(Imj);
%****************************************
L=wiener2(Imj,[3  3]);
% figure(1);imshow(L);
%****************************************

I=L;
[H_Image ,W_Image]=size(I);
mid=round(H_Image/2);
NewMin =0;

%% Fuzzification
for i=1 : H_Image
for j=1 : W_Image
if ((I(i,j) >= 0)&& (I(i,j)<= NewMin ))
            I (i,j)=0;
elseif (((I(i,j) >=NewMin)&& (I(i,j)<= mid)))
            I(i,j)=((1/(mid- NewMin)*NewMin) + (1/((mid-NewMin)*
I(i,j))));
end
if (((I(i,j) >= mid) && (I(i,j)<=H_Image)))
            I(i,j)=(1/(((H_Image- mid)*mid) + (1/(H_Image-mid)*I(i,j))));
end
if  ((I(i,j) >=H_Image) && (I(i,j)<= 255 ))
            I (i,j)=1;
end

end
end
```

```
%% you can change image input
% A=imread('01-O-01.bmp');
ImageResult=I;
ImageResult=im2double(ImageResult);
%I=double(I);
% A=double(A);
[ImageHeight ImageWidth]=size(I);

%% structuring element as Mask Matrix
B=[0.3 0.8 0.3; 0.8 1 0.8; 0.3 0.8 0.3];

%% You can Modify Structuring Element of B
% for example like below
% B=[1 150 125 150 1 ; 125 150 200 150 125; 125 150 200 150 125; 125 150
200 150 125;1 150 125 150 1];

%% it is necessary to identify Mask height and width
[MaskHeight MaskWidth]=size(B);

%% compute start iteration
Start=round(MaskHeight/2);

%% compute the last iteration
Last=fix(MaskHeight/2);

%% image iteration
for j=Start:ImageHeight-Last
for k=Start:ImageWidth-Last
        %% capture image corresponding to the structuring element
        PartOfMatrixA=I(j-Last:j+Last, k-Last:k+Last);
        Index=0;
        %% Masiking Iteration
for m=1:MaskHeight
for n=1:MaskHeight
                Index=Index+1;
                %% compare Matrix A with Structuring element B
                %% And put to DummyMin(Index)
if (PartOfMatrixA(m,n)>B(m,n))
                    DummyMin(Index)=PartOfMatrixA(m,n);
else
                    DummyMin(Index)=B(m,n);
end
end
end

        %% Select max value from the minum values obtained of
DummyMin(Index)
        ImageResult(j,k)= min(DummyMin);
end
end

%% Display the original image and result
% imshow([uint8(A) ImageResult]);
K=ImageResult;
% %**************************************
```

```
% %Defuzzification
for i=1 : H_Image
for j=1 : W_Image
if ((K(i,j) >= 0) &&(K(i,j) <= NewMin))
            J(i,j)= K(i,j);
end
if  ((K(i,j) >= NewMin) &&(K(i,j) <= mid))
            J(i,j)=(mid-NewMin)*(K(i,j)+NewMin);
end
if  ((K(i,j) >= mid) &&(K(i,j) <= H_Image));
            J(i,j)=(H_Image-mid)*(K(i,j)+mid);
end
end
end
% figure (2);
% subplot(1,2,1);
% imshow(Imj);title('(a)Original image');
% subplot(1,2,2);
% imshow(J);title('(b) Dilation image');

% %*******************************************
% L=im2bw(L, 0.5);
% J=im2bw(J, 0.5);
Imj=double(Imj);
D= and (Imj,J);

%D=double(D);
% figure (3);
% subplot(1,2,1);
% imshow(Imj);title('(a)Original image');
% subplot(1,2,2);
% imshow(D);title('(b) Enhanced image');
newmiddle=[front '-OO-' last  '.bmp'];
newfilename=[folderdestination  newmiddle];
imwrite(D,newfilename);
waitbar(numberof/(71*8));
end
end
close (h);
```

## Step 2:

```
clear all;close all;
h=waitbar (0,'please wait 5 minute ....');
folderdestination = ['D:\DISK1\DB_ITS_2009\OILY\Dilation2\'];
numberof=0;
for x=1:71
if x<10
        front=['0' num2str(x)];
else
        front=[num2str(x)];
end
for y=1:8
       numberof=numberof+1;
if y<10
        last=['0' num2str(y)];
else
        last=[num2str(y)];
```

```
end
    filename=[front '-OO-' last '.bmp'];
Imj=imread(filename);
 L=double(Imj);
%***************************************

% %*******************************************
%J=double (J);
invimg=1 - L;% there the inverse for dilated image
invimg=double(invimg);
% figure (3);
% subplot(1,2,1);
% imshow(Imj);title('(a)Dilated image');
% subplot(1,2,2);
% imshow(invimg);title('(b) Inverse of Dilated image');
newmiddle=[front '-VV-' last  '.bmp'];
newfilename=[folderdestination  newmiddle];
imwrite(invimg,newfilename);
waitbar(numberof/(71*8));
end
end
close (h);
```

## Step 3:

```
clear all;close all;
% h=waitbar (0,'please wait 5 minute ....');
% folderdestination = ['D:\DISK1\DB_ITS_2009\OILY\Erosion\'];
% numberof=0;
% for x=1:71
%     if x<10
%         front=['0' num2str(x)];
%     else
%         front=[num2str(x)];
%     end
%     for y=1:8
%        numberof=numberof+1;
%     if y<10
%         last=['0' num2str(y)];
%     else
%         last=[num2str(y)];
%     end
%     filename=[front '-O-' last '.bmp'];
Imj=imread('12-O-01.bmp');
L=double(Imj);

%***************************************
I=L;
[H_Image ,W_Image]=size(I);
mid=round(H_Image/2);
NewMin =0;

%% Fuzzification
for i=1 : H_Image
for j=1 : W_Image
if ((I(i,j) >= 0)&& (I(i,j)<= NewMin ))
            I (i,j)=0;
elseif (((I(i,j) >=NewMin)&& (I(i,j)<= mid)))
```

```
            I(i,j)=((1/(mid- NewMin)*NewMin) + (1/((mid-NewMin)*
I(i,j))));
end
if (((I(i,j) >= mid) && (I(i,j)<=H_Image)))
            I(i,j)=(1/(((H_Image- mid)*mid) + (1/(H_Image-mid)*I(i,j))));
end
if  ((I(i,j) >=H_Image) && (I(i,j)<= 255 ))
            I (i,j)=1;
end

end
end

%% you can change image input
% A=imread('01-O-01.bmp');
ImageResult=I;
ImageResult=double(ImageResult);

[ImageHeight ImageWidth]=size(I);

%% structuring element as Mask Matrix
B=[0.3 0.8 0.3; 0.8 1 0.8; 0.3 0.8 0.3];

%% You can Modify Structuring Element of B
% for example like below
% B=[1 150 125 150 1 ; 125 150 200 150 125; 125 150 200 150 125; 125 150
200 150 125;1 150 125 150 1];

%% it is necessary to identify Mask height and width
[MaskHeight MaskWidth]=size(B);

%% compute start iteration
Start=round(MaskHeight/2);

%% compute the last iteration
Last=fix(MaskHeight/2);

%% image iteration
for j=Start:ImageHeight-Last
for k=Start:ImageWidth-Last
        %% capture image corresponding to the structuring element
        PartOfMatrixA=I(j-Last:j+Last, k-Last:k+Last);
        Index=0;
        %% Masiking Iteration
for m=1:MaskHeight
for n=1:MaskHeight
                Index=Index+1;
                %% compare Matrix A with Structuring element B
                %% And put to DummyMin(Index)

if ((PartOfMatrixA(m,n))> B(m,n))
                    DummyMin(Index)=PartOfMatrixA(m,n);
else
                    DummyMin(Index)=B(m,n);
end
```

```
end
end

        %% Select max value from the minum values obtained of
DummyMin(Index)
        ImageResult(j,k)= min (DummyMin);
end
end
ImageResult=double(ImageResult);
%% Display the original image and result
% imshow([uint8(A) ImageResult]);
K=ImageResult;

% %**************************************
% %Defuzzification
for i=1 : H_Image
for j=1 : W_Image
if ((K(i,j) >= 0) &&(K(i,j) <= NewMin))
            J(i,j)= K(i,j);
end
if  ((K(i,j) >= NewMin) &&(K(i,j) <= mid))
            J(i,j)=(mid-NewMin)*(K(i,j)+NewMin);
end
if  ((K(i,j) >= mid) &&(K(i,j) <= H_Image));
            J(i,j)=(H_Image-mid)*(K(i,j)+mid);
end
end
end
J=double(J);
figure (2);
subplot(1,2,1);
imshow(Imj);title('(a)Original image');
subplot(1,2,2);
imshow(J);title('(b) Erosion image');

% %******************************************
% newmiddle=[front '-EE-' last  '.bmp'];
% newfilename=[folderdestination  newmiddle];
% imwrite(J,newfilename);
% waitbar(numberof/(71*8));
%     end
% end
% close (h);
```

## Step 4:

```
clear all;close all;
h=waitbar (0,'please wait 5 minute ....');
folderdestination = ['D:\DISK1\DB_ITS_2009\OILY\int_ero_inv\'];
numberof=0;
for x=1:71
if x<10
        front=['0' num2str(x)];
else
        front=[num2str(x)];
end
for y=1:8
       numberof=numberof+1;
```

```
if y<10
        last=['0' num2str(y)];
else
        last=[num2str(y)];
end
    filename1=[front '-VV-' last '.bmp'];
    filename2=[front '-EE-' last '.bmp'];
Imjv=imread(filename1);
Imje=imread(filename2);
L1=double(Imjv);
L2=double(Imje);
L2=im2bw(L2, 0.5);

D= and (L1,L2);
D=double(D);
%****************************************
% figure (1);
% subplot(1,3,1);
% imshow(Imjv);title('(A)Inverse of Dilated image');
% subplot(1,3,2);
% imshow(Imje);title('(B) Erosion image');
% subplot(1,3,3);
% imshow(D);title('(C) Intersection of A&B images');

% %*********************************************
newmiddle=[front '-INT-' last  '.bmp'];
newfilename=[folderdestination  newmiddle];
imwrite(D,newfilename);
waitbar(numberof/(71*8));
end
end
close (h);
```

## Step5:

```
clear all;close all;
h=waitbar (0,'please wait 5 minute ....');
folderdestination = ['D:\DISK1\DB_ITS_2009\OILY\final_Result\'];
numberof=0;
for x=1:71
if x<10
        front=['0' num2str(x)];
else
        front=[num2str(x)];
end
for y=1:8
       numberof=numberof+1;
if y<10
        last=['0' num2str(y)];
else
        last=[num2str(y)];
end
    filename1=[front '-OO-' last '.bmp'];
    filename2=[front '-INT-' last '.bmp'];
Imjv=imread(filename1);
Imje=imread(filename2);
L1=double(Imjv);
L2=double(Imje);
```

```
D= or (L1,L2);
D=double(D);
%****************************************
% figure (1);
% subplot(1,3,1);
% imshow(Imjv);title('(A) Dilated image');
% subplot(1,3,2);
% imshow(Imje);title('(B) Intersection Operation ');
% subplot(1,3,3);
% imshow(D);title('(C) Enhanced image');

% %*****************************************
newmiddle=[front '-EN-' last  '.bmp'];
newfilename=[folderdestination  newmiddle];
imwrite(D,newfilename);
waitbar(numberof/(71*8));
end
end
close (h);
```

## Appendix D

## The MALAB code For Performance Evaluation

```
clear all;close all;
h=waitbar (0,'please wait 5 minute ....');
%folderdestination = ['D:\DISK1\DB_ITS_2009\OILY\Old_method\Two&Three\'];
numberof=0;
for x=1:71
if x<10
        front=['0' num2str(x)];
else
        front=[num2str(x)];
end
    display('The FSIM for Person Number ');disp(x);
for y=1:8
      display('The FSIM for finger Number');disp(y);
       numberof=numberof+1;
if y<10
        last=['0' num2str(y)];
else
        last=[num2str(y)];
end
    filename1=[front '-old-' last '.bmp'];
    filename2=[front '-D-' last '.bmp'];
imageDis=imread(filename1);
imageRef=imread(filename2);
%***************************************
FeatureSIM(imageRef, imageDis);
% %*********************************************
%newmiddle=[front '-INT-' last  '.bmp'];
%newfilename=[folderdestination  newmiddle];
%imwrite(D,newfilename);
waitbar(numberof/(71*8));
end
end
close (h);
```

## Appendix E

### The GCS for Dry , Oily, and Neutral Fingerprint Image

## For Dry:

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0118 | 0.0117 | 0.0119 | 0.0122 | 0.0120 | 0.0118 | 0.0118 | 0.0119 |
| 2. | 0.0118 | 0.0119 | 0.0118 | 0.0119 | 0.0119 | 0.0119 | 0.0119 | 0.0119 |
| 3. | 0.0119 | 0.0117 | 0.0118 | 0.0118 | 0.0116 | 0.0117 | 0.0119 | 0.0117 |
| 4. | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0118 | 0.0118 | 0.0120 |
| 5. | 0.0118 | 0.0117 | 0.0117 | 0.0119 | 0.0117 | 0.0117 | 0.0118 | 0.0117 |
| 6. | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 |
| 7. | 0.0118 | 0.0117 | 0.0118 | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0118 |
| 8. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 9. | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0118 | 0.0117 |
| 10. | 0.0118 | 0.0118 | 0.0120 | 0.0119 | 0.0118 | 0.0120 | 0.0119 | 0.0118 |
| 11. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 12. | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0117 |
| 13. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0118 |
| 14. | 0.0119 | 0.0117 | 0.0118 | 0.0117 | 0.0116 | 0.0118 | 0.0118 | 0.0117 |
| 15. | 0.0119 | 0.0120 | 0.0120 | 0.0119 | 0.0120 | 0.0120 | 0.0118 | 0.0119 |
| 16. | 0.0116 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0118 | 0.0121 | 0.0119 |
| 17. | 0.0119 | 0.0121 | 0.0120 | 0.0119 | 0.0120 | 0.0121 | 0.0121 | 0.0119 |
| 18. | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0115 | 0.0117 | 0.0118 |
| 19. | 0.0123 | 0.0122 | 0.0121 | 0.0123 | 0.0124 | 0.0125 | 0.0122 | 0.0121 |
| 20. | 0.0121 | 0.0120 | 0.0119 | 0.0121 | 0.0121 | 0.0122 | 0.0121 | 0.0122 |
| 21. | 0.0117 | 0.0119 | 0.0118 | 0.0118 | 0.0116 | 0.0118 | 0.0117 | 0.0117 |
| 22. | 0.0121 | 0.0122 | 0.0123 | 0.0123 | 0.0121 | 0.0121 | 0.0122 | 0.0121 |
| 23. | 0.0116 | 0.0119 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0116 | 0.0119 |
| 24. | 0.0118 | 0.0118 | 0.0119 | 0.0119 | 0.0119 | 0.0121 | 0.0121 | 0.0120 |
| 25. | 0.0117 | 0.0118 | 0.0118 | 0.0120 | 0.0117 | 0.0117 | 0.0118 | 0.0117 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 26. | 0.0116 | 0.0118 | 0.0118 | 0.0118 | 0.0118 | 0.0117 | 0.0117 | 0.0117 |
| 27. | 0.0119 | 0.0117 | 0.0118 | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0117 |
| 28. | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0116 |
| 29. | 0.0117 | 0.0118 | 0.0120 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0118 |
| 30. | 0.0117 | 0.0118 | 0.0117 | 0.0116 | 0.0116 | 0.0118 | 0.0117 | 0.0117 |
| 31. | 0.0116 | 0.0118 | 0.0116 | 0.0119 | 0.0117 | 0.0118 | 0.0116 | 0.0117 |
| 32. | 0.0116 | 0.0119 | 0.0116 | 0.0120 | 0.0119 | 0.0117 | 0.0116 | 0.0118 |
| 33. | 0.0117 | 0.0119 | 0.0119 | 0.0118 | 0.0117 | 0.0118 | 0.0118 | 0.0117 |
| 34. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0116 |
| 35. | 0.0119 | 0.0118 | 0.0117 | 0.0118 | 0.0119 | 0.0116 | 0.0119 | 0.0116 |
| 36. | 0.0116 | 0.0117 | 0.0117 | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 |
| 37. | 0.0118 | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0117 | 0.0116 |
| 38. | 0.0120 | 0.0118 | 0.0119 | 0.0118 | 0.0119 | 0.0117 | 0.0118 | 0.0119 |
| 39. | 0.0116 | 0.0121 | 0.0118 | 0.0117 | 0.0118 | 0.0119 | 0.0117 | 0.0117 |
| 40. | 0.0121 | 0.0118 | 0.0120 | 0.0119 | 0.0117 | 0.0118 | 0.0117 | 0.0117 |
| 41. | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0117 |
| 42. | 0.0120 | 0.0118 | 0.0117 | 0.0119 | 0.0119 | 0.0119 | 0.0118 | 0.0118 |
| 43. | 0.0120 | 0.0120 | 0.0120 | 0.0118 | 0.0120 | 0.0120 | 0.0120 | 0.0118 |
| 44. | 0.0117 | 0.0119 | 0.0117 | 0.0117 | 0.0116 | 0.0119 | 0.0117 | 0.0117 |
| 45. | 0.0123 | 0.0118 | 0.0119 | 0.0117 | 0.0119 | 0.0118 | 0.0117 | 0.0118 |
| 46. | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0118 | 0.0117 |
| 47. | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0117 | 0.0117 | 0.0117 |
| 48. | 0.0118 | 0.0119 | 0.0118 | 0.0119 | 0.0117 | 0.0117 | 0.0119 | 0.0119 |
| 49. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 50. | 0.0117 | 0.0117 | 0.0119 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 51. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0115 | 0.0117 | 0.0116 | 0.0117 |
| 52. | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0117 |
| 53. | 0.0119 | 0.0117 | 0.0119 | 0.0119 | 0.0116 | 0.0116 | 0.0118 | 0.0118 |
| 54. | 0.0117 | 0.0117 | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0119 |
| 55. | 0.0118 | 0.0118 | 0.0117 | 0.0117 | 0.0120 | 0.0117 | 0.0116 | 0.0118 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 56. | 0.0119 | 0.0121 | 0.0120 | 0.0119 | 0.0118 | 0.0120 | 0.0118 | 0.0120 |
| 57. | 0.0116 | 0.0116 | 0.0116 | 0.0118 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 58. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 59. | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 60. | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0116 |
| 61. | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0118 |
| 62. | 0.0119 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0117 |
| 63. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 64. | 0.0118 | 0.0119 | 0.0118 | 0.0118 | 0.0119 | 0.0119 | 0.0118 | 0.0117 |
| 65. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 |
| 66. | 0.0117 | 0.0116 | 0.0118 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 |
| 67. | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0116 |
| 68. | 0.0117 | 0.0118 | 0.0118 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0116 |
| 69. | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0117 | 0.0116 |
| 70. | 0.0118 | 0.0118 | 0.0119 | 0.0117 | 0.0117 | 0.0118 | 0.0119 | 0.0118 |
| 71. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 |

**For Neutral**

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0119 | 0.0118 | 0.0118 | 0.0120 | 0.0118 | 0.0119 | 0.0119 | 0.0119 |
| 2. | 0.0119 | 0.0117 | 0.0118 | 0.0119 | 0.0118 | 0.0119 | 0.0118 | 0.0117 |
| 3. | 0.0119 | 0.0116 | 0.0118 | 0.0120 | 0.0116 | 0.0115 | 0.0118 | 0.0117 |
| 4. | 0.0117 | 0.0119 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0117 |
| 5. | 0.0118 | 0.0117 | 0.0117 | 0.0120 | 0.0117 | 0.0116 | 0.0116 | 0.0118 |
| 6. | 0.0118 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0118 | 0.0116 | 0.0116 |
| 7. | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0118 | 0.0119 | 0.0119 | 0.0118 |
| 8. | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0116 |
| 9. | 0.0115 | 0.0115 | 0.0114 | 0.0115 | 0.0115 | 0.0114 | 0.0116 | 0.0115 |
| 10. | 0.0119 | 0.0118 | 0.0119 | 0.0120 | 0.0119 | 0.0119 | 0.0118 | 0.0119 |
| 11. | 0.0118 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0117 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 12. | 0.0118 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0116 | 0.0116 |
| 13. | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0117 | 0.0117 | 0.0116 | 0.0117 |
| 14. | 0.0120 | 0.0117 | 0.0116 | 0.0116 | 0.0115 | 0.0115 | 0.0115 | 0.0115 |
| 15. | 0.0121 | 0.0118 | 0.0120 | 0.0120 | 0.0119 | 0.0119 | 0.0117 | 0.0118 |
| 16. | 0.0119 | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 |
| 17. | 0.0120 | 0.0120 | 0.0120 | 0.0119 | 0.0119 | 0.0118 | 0.0119 | 0.0118 |
| 18. | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0118 | 0.0116 | 0.0117 | 0.0117 |
| 19. | 0.0118 | 0.0121 | 0.0120 | 0.0121 | 0.0121 | 0.0123 | 0.0121 | 0.0123 |
| 20. | 0.0122 | 0.0121 | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0121 |
| 21. | 0.0118 | 0.0118 | 0.0118 | 0.0120 | 0.0121 | 0.0120 | 0.0118 | 0.0117 |
| 22. | 0.0123 | 0.0120 | 0.0120 | 0.0120 | 0.0119 | 0.0120 | 0.0119 | 0.0119 |
| 23. | 0.0116 | 0.0116 | 0.0115 | 0.0115 | 0.0116 | 0.0116 | 0.0116 | 0.0118 |
| 24. | 0.0120 | 0.0119 | 0.0120 | 0.0121 | 0.0120 | 0.0122 | 0.0123 | 0.0119 |
| 25. | 0.0119 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0118 | 0.0117 | 0.0117 |
| 26. | 0.0118 | 0.0117 | 0.0117 | 0.0119 | 0.0119 | 0.0120 | 0.0117 | 0.0118 |
| 27. | 0.0116 | 0.0115 | 0.0116 | 0.0115 | 0.0116 | 0.0115 | 0.0115 | 0.0115 |
| 28. | 0.0116 | 0.0115 | 0.0117 | 0.0115 | 0.0115 | 0.0115 | 0.0115 | 0.0115 |
| 29. | 0.0118 | 0.0117 | 0.0118 | 0.0117 | 0.0118 | 0.0118 | 0.0118 | 0.0119 |
| 30. | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 |
| 31. | 0.0117 | 0.0117 | 0.0118 | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0117 |
| 32. | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0118 | 0.0120 |
| 33. | 0.0119 | 0.0117 | 0.0119 | 0.0117 | 0.0119 | 0.0118 | 0.0117 | 0.0117 |
| 34. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 |
| 35. | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0115 | 0.0116 |
| 36. | 0.0117 | 0.0119 | 0.0120 | 0.0118 | 0.0119 | 0.0118 | 0.0117 | 0.0118 |
| 37. | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0115 | 0.0116 | 0.0115 | 0.0115 |
| 38. | 0.0119 | 0.0118 | 0.0119 | 0.0118 | 0.0118 | 0.0117 | 0.0118 | 0.0120 |
| 39. | 0.0118 | 0.0118 | 0.0117 | 0.0116 | 0.0119 | 0.0116 | 0.0115 | 0.0115 |
| 40. | 0.0117 | 0.0121 | 0.0119 | 0.0119 | 0.0119 | 0.0119 | 0.0120 | 0.0119 |
| 41. | 0.0118 | 0.0120 | 0.0119 | 0.0120 | 0.0119 | 0.0119 | 0.0119 | 0.0118 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 42. | 0.0120 | 0.0117 | 0.0120 | 0.0120 | 0.0118 | 0.0118 | 0.0118 | 0.0118 |
| 43. | 0.0117 | 0.0116 | 0.0119 | 0.0120 | 0.0121 | 0.0120 | 0.0121 | 0.0121 |
| 44. | 0.0121 | 0.0120 | 0.0119 | 0.0121 | 0.0120 | 0.0119 | 0.0119 | 0.0120 |
| 45. | 0.0119 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0117 | 0.0118 | 0.0116 |
| 46. | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0118 | 0.0115 |
| 47. | 0.0118 | 0.0118 | 0.0119 | 0.0119 | 0.0118 | 0.0119 | 0.0117 | 0.0118 |
| 48. | 0.0119 | 0.0120 | 0.0119 | 0.0120 | 0.0119 | 0.0121 | 0.0120 | 0.0117 |
| 49. | 0.0116 | 0.0115 | 0.0115 | 0.0115 | 0.0118 | 0.0116 | 0.0115 | 0.0116 |
| 50. | 0.0117 | 0.0118 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 |
| 51. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0116 | 0.0115 | 0.0117 | 0.0116 |
| 52. | 0.0118 | 0.0116 | 0.0116 | 0.0116 | 0.0115 | 0.0116 | 0.0116 | 0.0116 |
| 53. | 0.0118 | 0.0119 | 0.0118 | 0.0117 | 0.0118 | 0.0116 | 0.0119 | 0.0118 |
| 54. | 0.0118 | 0.0118 | 0.0119 | 0.0118 | 0.0117 | 0.0120 | 0.0119 | 0.0119 |
| 55. | 0.0118 | 0.0120 | 0.0118 | 0.0121 | 0.0118 | 0.0119 | 0.0118 | 0.0119 |
| 56. | 0.0120 | 0.0119 | 0.0120 | 0.0120 | 0.0121 | 0.0118 | 0.0120 | 0.0120 |
| 57. | 0.0116 | 0.0116 | 0.0118 | 0.0117 | 0.0116 | 0.0116 | 0.0118 | 0.0116 |
| 58. | 0.0116 | 0.0115 | 0.0115 | 0.0115 | 0.0118 | 0.0116 | 0.0115 | 0.0116 |
| 59. | 0.0116 | 0.0116 | 0.0118 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 |
| 60. | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 |
| 61. | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0116 | 0.0115 | 0.0115 | 0.0115 |
| 62. | 0.0118 | 0.0118 | 0.0118 | 0.0118 | 0.0116 | 0.0116 | 0.0118 | 0.0118 |
| 63. | 0.0116 | 0.0117 | 0.0116 | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 |
| 64. | 0.0120 | 0.0118 | 0.0118 | 0.0120 | 0.0118 | 0.0119 | 0.0119 | 0.0119 |
| 65. | 0.0117 | 0.0118 | 0.0118 | 0.0117 | 0.0116 | 0.0117 | 0.0116 | 0.0116 |
| 66. | 0.0118 | 0.0116 | 0.0118 | 0.0116 | 0.0117 | 0.0116 | 0.0115 | 0.0115 |
| 67. | 0.0116 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0117 | 0.0118 | 0.0116 |
| 68. | 0.0119 | 0.0117 | 0.0117 | 0.0119 | 0.0120 | 0.0119 | 0.0116 | 0.0117 |
| 69. | 0.0117 | 0.0117 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0116 | 0.0117 |
| 70. | 0.0119 | 0.0118 | 0.0118 | 0.0117 | 0.0119 | 0.0118 | 0.0118 | 0.0117 |
| 71. | 0.0119 | 0.0118 | 0.0118 | 0.0118 | 0.0119 | 0.0119 | 0.0118 | 0.0119 |

**For Oily**

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0121 | 0.0126 | 0.0121 | 0.0121 | 0.0123 | 0.0125 | 0.0122 | 0.0123 |
| 2. | 0.0123 | 0.0123 | 0.0124 | 0.0122 | 0.0122 | 0.0121 | 0.0124 | 0.0122 |
| 3. | 0.0120 | 0.0122 | 0.0120 | 0.0121 | 0.0120 | 0.0123 | 0.0123 | 0.0121 |
| 4. | 0.0122 | 0.0122 | 0.0123 | 0.0121 | 0.0122 | 0.0122 | 0.0122 | 0.0121 |
| 5. | 0.0120 | 0.0121 | 0.0122 | 0.0121 | 0.0122 | 0.0120 | 0.0121 | 0.0120 |
| 6. | 0.0120 | 0.0122 | 0.0122 | 0.0122 | 0.0121 | 0.0120 | 0.0122 | 0.0123 |
| 7. | 0.0124 | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0121 |
| 8. | 0.0122 | 0.0122 | 0.0121 | 0.0122 | 0.0123 | 0.0122 | 0.0124 | 0.0124 |
| 9. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0121 |
| 10. | 0.0123 | 0.0122 | 0.0122 | 0.0121 | 0.0123 | 0.0123 | 0.0121 | 0.0121 |
| 11. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0120 | 0.0121 | 0.0123 | 0.0120 |
| 12. | 0.0120 | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0120 |
| 13. | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 |
| 14. | 0.0122 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0120 | 0.0120 | 0.0121 |
| 15. | 0.0127 | 0.0128 | 0.0126 | 0.0126 | 0.0127 | 0.0126 | 0.0123 | 0.0121 |
| 16. | 0.0121 | 0.0122 | 0.0122 | 0.0121 | 0.0121 | 0.0122 | 0.0122 | 0.0121 |
| 17. | 0.0124 | 0.0123 | 0.0121 | 0.0121 | 0.0124 | 0.0122 | 0.0123 | 0.0123 |
| 18. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0120 | 0.0120 | 0.0121 | 0.0120 |
| 19. | 0.0127 | 0.0125 | 0.0123 | 0.0127 | 0.0126 | 0.0128 | 0.0128 | 0.0127 |
| 20. | 0.0127 | 0.0126 | 0.0128 | 0.0126 | 0.0126 | 0.0127 | 0.0126 | 0.0129 |
| 21. | 0.0128 | 0.0127 | 0.0129 | 0.0125 | 0.0125 | 0.0125 | 0.0126 | 0.0124 |
| 22. | 0.0128 | 0.0130 | 0.0129 | 0.0125 | 0.0127 | 0.0125 | 0.0125 | 0.0125 |
| 23. | 0.0123 | 0.0122 | 0.0121 | 0.0121 | 0.0126 | 0.0122 | 0.0124 | 0.0122 |
| 24. | 0.0122 | 0.0125 | 0.0126 | 0.0124 | 0.0122 | 0.0122 | 0.0125 | 0.0123 |
| 25. | 0.0127 | 0.0124 | 0.0125 | 0.0125 | 0.0125 | 0.0125 | 0.0125 | 0.0125 |
| 26. | 0.0122 | 0.0121 | 0.0122 | 0.0120 | 0.0121 | 0.0122 | 0.0122 | 0.0121 |
| 27. | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0119 | 0.0120 | 0.0120 | 0.0121 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 28. | 0.0123 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0120 |
| 29. | 0.0125 | 0.0124 | 0.0123 | 0.0124 | 0.0123 | 0.0122 | 0.0123 | 0.0122 |
| 30. | 0.0127 | 0.0127 | 0.0125 | 0.0125 | 0.0125 | 0.0124 | 0.0126 | 0.0123 |
| 31. | 0.0122 | 0.0125 | 0.0121 | 0.0123 | 0.0124 | 0.0123 | 0.0124 | 0.0124 |
| 32. | 0.0123 | 0.0121 | 0.0122 | 0.0123 | 0.0122 | 0.0123 | 0.0122 | 0.0122 |
| 33. | 0.0123 | 0.0128 | 0.0121 | 0.0122 | 0.0124 | 0.0124 | 0.0122 | 0.0121 |
| 34. | 0.0123 | 0.0124 | 0.0123 | 0.0123 | 0.0122 | 0.0122 | 0.0121 | 0.0121 |
| 35. | 0.0123 | 0.0122 | 0.0123 | 0.0124 | 0.0122 | 0.0121 | 0.0121 | 0.0120 |
| 36. | 0.0124 | 0.0121 | 0.0120 | 0.0125 | 0.0126 | 0.0123 | 0.0121 | 0.0121 |
| 37. | 0.0121 | 0.0120 | 0.0120 | 0.0120 | 0.0121 | 0.0123 | 0.0121 | 0.0122 |
| 38. | 0.0123 | 0.0122 | 0.0124 | 0.0125 | 0.0125 | 0.0123 | 0.0122 | 0.0123 |
| 39. | 0.0122 | 0.0123 | 0.0124 | 0.0122 | 0.0124 | 0.0123 | 0.0123 | 0.0122 |
| 40. | 0.0128 | 0.0127 | 0.0127 | 0.0122 | 0.0122 | 0.0130 | 0.0125 | 0.0124 |
| 41. | 0.0125 | 0.0124 | 0.0123 | 0.0122 | 0.0123 | 0.0126 | 0.0127 | 0.0125 |
| 42. | 0.0126 | 0.0123 | 0.0125 | 0.0121 | 0.0124 | 0.0125 | 0.0125 | 0.0127 |
| 43. | 0.0128 | 0.0123 | 0.0122 | 0.0126 | 0.0122 | 0.0127 | 0.0124 | 0.0126 |
| 44. | 0.0122 | 0.0121 | 0.0124 | 0.0123 | 0.0121 | 0.0120 | 0.0120 | 0.0122 |
| 45. | 0.0121 | 0.0124 | 0.0121 | 0.0121 | 0.0121 | 0.0120 | 0.0121 | 0.0121 |
| 46. | 0.0126 | 0.0124 | 0.0124 | 0.0124 | 0.0127 | 0.0124 | 0.0124 | 0.0126 |
| 47. | 0.0122 | 0.0123 | 0.0125 | 0.0123 | 0.0125 | 0.0120 | 0.0121 | 0.0120 |
| 48. | 0.0123 | 0.0122 | 0.0123 | 0.0124 | 0.0123 | 0.0123 | 0.0123 | 0.0124 |
| 49. | 0.0125 | 0.0124 | 0.0121 | 0.0123 | 0.0124 | 0.0121 | 0.0124 | 0.0124 |
| 50. | 0.0122 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0121 | 0.0122 |
| 51. | 0.0122 | 0.0121 | 0.0121 | 0.0122 | 0.0120 | 0.0120 | 0.0120 | 0.0120 |
| 52. | 0.0122 | 0.0121 | 0.0121 | 0.0123 | 0.0123 | 0.0123 | 0.0122 | 0.0120 |
| 53. | 0.0122 | 0.0123 | 0.0125 | 0.0121 | 0.0122 | 0.0123 | 0.0122 | 0.0121 |
| 54. | 0.0124 | 0.0124 | 0.0124 | 0.0124 | 0.0124 | 0.0126 | 0.0125 | 0.0126 |
| 55. | 0.0124 | 0.0124 | 0.0125 | 0.0124 | 0.0128 | 0.0126 | 0.0123 | 0.0123 |
| 56. | 0.0127 | 0.0125 | 0.0126 | 0.0124 | 0.0124 | 0.0124 | 0.0124 | 0.0125 |
| 57. | 0.0125 | 0.0120 | 0.0121 | 0.0121 | 0.0121 | 0.0124 | 0.0124 | 0.0121 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 58. | 0.0125 | 0.0124 | 0.0121 | 0.0123 | 0.0124 | 0.0121 | 0.0124 | 0.0124 |
| 59. | 0.0122 | 0.0120 | 0.0121 | 0.0120 | 0.0120 | 0.0121 | 0.0121 | 0.0121 |
| 60. | 0.0121 | 0.0122 | 0.0123 | 0.0119 | 0.0121 | 0.0124 | 0.0120 | 0.0120 |
| 61. | 0.0123 | 0.0123 | 0.0122 | 0.0120 | 0.0120 | 0.0120 | 0.0120 | 0.0121 |
| 62. | 0.0122 | 0.0124 | 0.0121 | 0.0121 | 0.0122 | 0.0121 | 0.0121 | 0.0122 |
| 63. | 0.0121 | 0.0120 | 0.0120 | 0.0123 | 0.0121 | 0.0123 | 0.0123 | 0.0123 |
| 64. | 0.0123 | 0.0122 | 0.0122 | 0.0123 | 0.0126 | 0.0121 | 0.0124 | 0.0122 |
| 65. | 0.0122 | 0.0122 | 0.0124 | 0.0121 | 0.0123 | 0.0123 | 0.0122 | 0.0122 |
| 66. | 0.0122 | 0.0122 | 0.0121 | 0.0123 | 0.0121 | 0.0120 | 0.0120 | 0.0120 |
| 67. | 0.0124 | 0.0122 | 0.0122 | 0.0122 | 0.0123 | 0.0126 | 0.0124 | 0.0122 |
| 68. | 0.0121 | 0.0121 | 0.0123 | 0.0121 | 0.0122 | 0.0122 | 0.0122 | 0.0122 |
| 69. | 0.0121 | 0.0121 | 0.0123 | 0.0121 | 0.0123 | 0.0121 | 0.0121 | 0.0121 |
| 70. | 0.0123 | 0.0125 | 0.0125 | 0.0124 | 0.0124 | 0.0122 | 0.0123 | 0.0125 |
| 71. | 0.0120 | 0.0121 | 0.0121 | 0.0120 | 0.0121 | 0.0121 | 0.0121 | 0.0121 |

## Appendix F

### The LCS for Dry , Oily, and Neutral Fingerprint Image

## For Dry:

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0129 | 0.0133 | 0.0126 | 0.0124 | 0.0158 | 0.0127 | 0.0119 | 0.0152 |
| 2. | 0.0127 | 0.0129 | 0.0126 | 0.0144 | 0.0127 | 0.0126 | 0.0138 | 0.0127 |
| 3. | 0.0130 | 0.0127 | 0.0120 | 0.0124 | 0.0129 | 0.0126 | 0.0138 | 0.0126 |
| 4. | 0.0132 | 0.0132 | 0.0132 | 0.0133 | 0.0132 | 0.0134 | 0.0130 | 0.0126 |
| 5. | 0.0128 | 0.0132 | 0.0127 | 0.0132 | 0.0129 | 0.0128 | 0.0127 | 0.0130 |
| 6. | 0.0136 | 0.0133 | 0.0133 | 0.0133 | 0.0134 | 0.0137 | 0.0131 | 0.0134 |
| 7. | 0.0127 | 0.0129 | 0.0122 | 0.0123 | 0.0129 | 0.0130 | 0.0128 | 0.0137 |
| 8. | 0.0132 | 0.0134 | 0.0136 | 0.0133 | 0.0133 | 0.0133 | 0.0135 | 0.0134 |
| 9. | 0.0125 | 0.0129 | 0.0130 | 0.0131 | 0.0128 | 0.0135 | 0.0126 | 0.0129 |
| 10. | 0.0128 | 0.0130 | 0.0125 | 0.0127 | 0.0130 | 0.0127 | 0.0128 | 0.0130 |
| 11. | 0.0135 | 0.0136 | 0.0134 | 0.0132 | 0.0134 | 0.0133 | 0.0132 | 0.0131 |
| 12. | 0.0131 | 0.0131 | 0.0130 | 0.0128 | 0.0130 | 0.0130 | 0.0131 | 0.0129 |
| 13. | 0.0128 | 0.0127 | 0.0129 | 0.0131 | 0.0128 | 0.0130 | 0.0128 | 0.0139 |
| 14. | 0.0130 | 0.0129 | 0.0129 | 0.0131 | 0.0132 | 0.0130 | 0.0129 | 0.0130 |
| 15. | 0.0168 | 0.0141 | 0.0141 | 0.0153 | 0.0137 | 0.0153 | 0.0140 | 0.0160 |
| 16. | 0.0131 | 0.0130 | 0.0131 | 0.0129 | 0.0131 | 0.0130 | 0.0171 | 0.0129 |
| 17. | 0.0130 | 0.0128 | 0.0139 | 0.0139 | 0.178 | 0.0151 | 0.0130 | 0.0137 |
| 18. | 0.0130 | 0.0123 | 0.0129 | 0.0127 | 0.0129 | 0.0134 | 0.0131 | 0.0129 |
| 19. | 0.0131 | 0.0163 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0131 |
| 20. | 0.0132 | 0.0129 | 0.0131 | 0.0131 | 0.0144 | 0.0132 | 0.0132 | 0.0132 |
| 21. | 0.0132 | 0.0176 | 0.0126 | 0.0124 | 0.0131 | 0.0129 | 0.0127 | 0.0139 |
| 22. | 0.0161 | 0.0165 | 0.0161 | 0.0161 | 0.0172 | 0.0161 | 0.0161 | 0.0184 |
| 23. | 0.0131 | 0.0173 | 0.0132 | 0.0133 | 0.0140 | 0.0132 | 0.0132 | 0.0157 |
| 24. | 0.0150 | 0.0130 | 0.0150 | 0.0133 | 0.0133 | 0.0218 | 0.0209 | 0.0213 |
| 25. | 0.0131 | 0.0131 | 0.0130 | 0.0124 | 0.0134 | 0.0131 | 0.0130 | 0.0131 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 26. | 0.0135 | 0.0133 | 0.0130 | 0.0132 | 0.0133 | 0.0134 | 0.0134 | 0.0137 |
| 27. | 0.0125 | 0.0132 | 0.0129 | 0.0128 | 0.0131 | 0.0129 | 0.0131 | 0.0131 |
| 28. | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0131 | 0.0129 | 0.0132 | 0.0131 |
| 29. | 0.0131 | 0.0128 | 0.0125 | 0.0130 | 0.0128 | 0.0129 | 0.0129 | 0.0127 |
| 30. | 0.0130 | 0.0132 | 0.0128 | 0.0129 | 0.0129 | 0.0127 | 0.0130 | 0.0129 |
| 31. | 0.0135 | 0.0134 | 0.0134 | 0.0128 | 0.0136 | 0.0132 | 0.0133 | 0.0134 |
| 32. | 0.0132 | 0.0127 | 0.0132 | 0.0128 | 0.0130 | 0.0132 | 0.0133 | 0.0129 |
| 33. | 0.0135 | 0.0149 | 0.0131 | 0.0129 | 0.0133 | 0.0125 | 0.0126 | 0.0133 |
| 34. | 0.0129 | 0.0131 | 0.0132 | 0.0131 | 0.0132 | 0.0132 | 0.0129 | 0.0131 |
| 35. | 0.0131 | 0.0128 | 0.0131 | 0.0131 | 0.0129 | 0.0131 | 0.0129 | 0.0132 |
| 36. | 0.0133 | 0.0131 | 0.0130 | 0.0133 | 0.0132 | 0.0130 | 0.0135 | 0.0135 |
| 37. | 0.0129 | 0.0130 | 0.0131 | 0.0131 | 0.0131 | 0.0129 | 0.0130 | 0.0130 |
| 38. | 0.0199 | 0.0125 | 0.0221 | 0.0157 | 0.0122 | 0.0124 | 0.0129 | 0.0153 |
| 39. | 0.0131 | 0.0181 | 0.0132 | 0.0131 | 0.0129 | 0.0137 | 0.0131 | 0.0131 |
| 40. | 0.0161 | 0.0129 | 0.0129 | 0.0128 | 0.0131 | 0.0129 | 0.0132 | 0.0132 |
| 41. | 0.0130 | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0130 | 0.0131 |
| 42. | 0.0138 | 0.0134 | 0.0134 | 0.0133 | 0.0126 | 0.0128 | 0.0131 | 0.0131 |
| 43. | 0.0173 | 0.0180 | 0.0133 | 0.0133 | 0.0185 | 0.0146 | 0.0150 | 0.0128 |
| 44. | 0.0129 | 0.0124 | 0.0128 | 0.0129 | 0.0131 | 0.0135 | 0.0129 | 0.0133 |
| 45. | 0.0131 | 0.0131 | 0.0175 | 0.0130 | 0.0143 | 0.0134 | 0.0132 | 0.0132 |
| 46. | 0.0134 | 0.0131 | 0.0131 | 0.0132 | 0.0131 | 0.0131 | 0.0128 | 0.0129 |
| 47. | 0.0130 | 0.0131 | 0.0131 | 0.0129 | 0.0131 | 0.0133 | 0.0130 | 0.0130 |
| 48. | 0.0132 | 0.0138 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0133 | 0.0137 |
| 49. | 0.0133 | 0.0133 | 0.0132 | 0.0133 | 0.0131 | 0.0132 | 0.0135 | 0.0132 |
| 50. | 0.0131 | 0.0130 | 0.0132 | 0.0131 | 0.0132 | 0.0131 | 0.0132 | 0.0132 |
| 51. | 0.0133 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0128 | 0.0130 | 0.0130 |
| 52. | 0.0132 | 0.0130 | 0.0132 | 0.0132 | 0.0131 | 0.0132 | 0.0132 | 0.0131 |
| 53. | 0.0134 | 0.0134 | 0.0134 | 0.0133 | 0.0132 | 0.0134 | 0.0132 | 0.0132 |
| 54. | 0.0130 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0132 | 0.0131 | 0.0144 |
| 55. | 0.0133 | 0.0131 | 0.0131 | 0.0130 | 0.0127 | 0.0134 | 0.0132 | 0.0131 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 56. | 0.0142 | 0.0132 | 0.0131 | 0.0176 | 0.0131 | 0.0144 | 0.0131 | 0.0129 |
| 57. | 0.0134 | 0.0135 | 0.0132 | 0.0132 | 0.0133 | 0.0135 | 0.0136 | 0.0134 |
| 58. | 0.0133 | 0.0133 | 0.0132 | 0.0133 | 0.0131 | 0.0132 | 0.0135 | 0.0132 |
| 59. | 0.0133 | 0.0134 | 0.0136 | 0.0131 | 0.0134 | 0.0133 | 0.0134 | 0.0135 |
| 60. | 0.0130 | 0.0137 | 0.0138 | 0.0136 | 0.0134 | 0.0135 | 0.0129 | 0.0136 |
| 61. | 0.0139 | 0.0132 | 0.0131 | 0.0129 | 0.0131 | 0.0131 | 0.0130 | 0.0126 |
| 62. | 0.0160 | 0.0132 | 0.0132 | 0.0132 | 0.0131 | 0.0131 | 0.0132 | 0.0133 |
| 63. | 0.0131 | 0.0130 | 0.0131 | 0.0131 | 0.0130 | 0.0132 | 0.0132 | 0.0132 |
| 64. | 0.0127 | 0.0129 | 0.0128 | 0.0129 | 0.0128 | 0.0128 | 0.0130 | 0.0129 |
| 65. | 0.0132 | 0.0133 | 0.0135 | 0.0134 | 0.0134 | 0.0133 | 0.0131 | 0.0136 |
| 66. | 0.0131 | 0.0131 | 0.0130 | 0.0131 | 0.0130 | 0.0132 | 0.0131 | 0.0131 |
| 67. | 0.0132 | 0.0131 | 0.0131 | 0.0131 | 0.0132 | 0.0131 | 0.0131 | 0.0131 |
| 68. | 0.0131 | 0.0129 | 0.0130 | 0.0135 | 0.0129 | 0.0148 | 0.0131 | 0.0131 |
| 69. | 0.0132 | 0.0132 | 0.0131 | 0.0133 | 0.0134 | 0.0133 | 0.0136 | 0.0133 |
| 70. | 0.0132 | 0.0129 | 0.0127 | 0.0130 | 0.0131 | 0.0138 | 0.0133 | 0.0130 |
| 71. | 0.0129 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0132 | 0.0131 | 0.0131 |

**For Neutral**

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0124 | 0.0125 | 0.0129 | 0.0165 | 0.0129 | 0.0133 | 0.0123 | 0.0132 |
| 2. | 0.0129 | 0.0131 | 0.0128 | 0.0128 | 0.0131 | 0.0130 | 0.0129 | 0.0132 |
| 3. | 0.0131 | 0.0132 | 0.0131 | 0.0189 | 0.0134 | 0.0134 | 0.0174 | 0.0132 |
| 4. | 0.0126 | 0.0128 | 0.0129 | 0.0127 | 0.0129 | 0.0127 | 0.0130 | 0.0128 |
| 5. | 0.0129 | 0.0129 | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0126 |
| 6. | 0.0132 | 0.0132 | 0.0128 | 0.0134 | 0.0132 | 0.0127 | 0.0133 | 0.0132 |
| 7. | 0.0135 | 0.0117 | 0.0134 | 0.0135 | 0.0122 | 0.0126 | 0.0135 | 0.0126 |
| 8. | 0.0131 | 0.0132 | 0.0131 | 0.0132 | 0.0131 | 0.0131 | 0.0131 | 0.0130 |
| 9. | 0.0134 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0133 | 0.0133 | 0.0132 |
| 10. | 0.0125 | 0.0129 | 0.0130 | 0.0124 | 0.0128 | 0.0131 | 0.0127 | 0.0126 |
| 11. | 0.0133 | 0.0133 | 0.0131 | 0.0130 | 0.0131 | 0.0133 | 0.0130 | 0.0130 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 12. | 0.0131 | 0.0128 | 0.0130 | 0.0131 | 0.0128 | 0.0129 | 0.0130 | 0.0131 |
| 13. | 0.0132 | 0.0131 | 0.0130 | 0.0131 | 0.0130 | 0.0129 | 0.0132 | 0.0130 |
| 14. | 0.0163 | 0.0132 | 0.0132 | 0.0133 | 0.0134 | 0.0131 | 0.0131 | 0.0130 |
| 15. | 0.0135 | 0.0134 | 0.0139 | 0.0165 | 0.0140 | 0.0134 | 0.0132 | 0.0170 |
| 16. | 0.0132 | 0.0132 | 0.0129 | 0.0131 | 0.0130 | 0.0131 | 0.0135 | 0.0131 |
| 17. | 0.0199 | 0.0155 | 0.0174 | 0.0151 | 0.0129 | 0.0130 | 0.0155 | 0.0136 |
| 18. | 0.0129 | 0.0129 | 0.0124 | 0.0128 | 0.0125 | 0.0129 | 0.0128 | 0.0128 |
| 19. | 0.0144 | 0.0133 | 0.0136 | 0.0142 | 0.0136 | 0.0133 | 0.0135 | 0.0132 |
| 20. | 0.0132 | 0.0132 | 0.0130 | 0.0131 | 0.0132 | 0.0132 | 0.0142 | 0.0173 |
| 21. | 0.0127 | 0.0126 | 0.0147 | 0.0186 | 0.0127 | 0.0184 | 0.0149 | 0.0130 |
| 22. | 0.0162 | 0.0149 | 0.0128 | 0.0129 | 0.0131 | 0.0129 | 0.0162 | 0.0151 |
| 23. | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0138 |
| 24. | 0.0159 | 0.0171 | 0.0126 | 0.0126 | 0.0190 | 0.0201 | 0.0126 | 0.0126 |
| 25. | 0.0125 | 0.0131 | 0.0130 | 0.0130 | 0.0131 | 0.0129 | 0.0132 | 0.0133 |
| 26. | 0.0129 | 0.0132 | 0.0130 | 0.0126 | 0.0135 | 0.0134 | 0.0132 | 0.0130 |
| 27. | 0.0131 | 0.0132 | 0.0131 | 0.0131 | 0.0131 | 0.0131 | 0.0133 | 0.0131 |
| 28. | 0.0130 | 0.0133 | 0.0130 | 0.0133 | 0.0135 | 0.0134 | 0.0132 | 0.0135 |
| 29. | 0.0126 | 0.0127 | 0.0128 | 0.0129 | 0.0129 | 0.0131 | 0.0133 | 0.0132 |
| 30. | 0.0128 | 0.0129 | 0.0126 | 0.0129 | 0.0129 | 0.0129 | 0.0130 | 0.0128 |
| 31. | 0.0130 | 0.0139 | 0.0130 | 0.0133 | 0.0132 | 0.0131 | 0.0130 | 0.0131 |
| 32. | 0.0130 | 0.0131 | 0.0131 | 0.0129 | 0.0130 | 0.0131 | 0.0130 | 0.0142 |
| 33. | 0.0131 | 0.0130 | 0.0127 | 0.0132 | 0.0135 | 0.0133 | 0.0133 | 0.0134 |
| 34. | 0.0132 | 0.0131 | 0.0134 | 0.0131 | 0.0129 | 0.0131 | 0.0130 | 0.0129 |
| 35. | 0.0129 | 0.0133 | 0.0131 | 0.0132 | 0.0131 | 0.0129 | 0.0131 | 0.0132 |
| 36. | 0.0131 | 0.0130 | 0.0200 | 0.0134 | 0.0129 | 0.0139 | 0.0131 | 0.0130 |
| 37. | 0.0132 | 0.0132 | 0.0130 | 0.0132 | 0.0133 | 0.0132 | 0.0133 | 0.0132 |
| 38. | 0.0175 | 0.0128 | 0.0150 | 0.0135 | 0.0132 | 0.0133 | 0.0124 | 0.0191 |
| 39. | 0.0128 | 0.0127 | 0.0130 | 0.0131 | 0.0190 | 0.0131 | 0.0132 | 0.0132 |
| 40. | 0.0132 | 0.0132 | 0.0141 | 0.0139 | 0.0143 | 0.0134 | 0.0161 | 0.0136 |
| 41. | 0.0127 | 0.0181 | 0.0127 | 0.0177 | 0.0127 | 0.0172 | 0.0127 | 0.0128 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 42. | 0.0191 | 0.0126 | 0.0132 | 0.0154 | 0.0126 | 0.0126 | 0.0127 | 0.0126 |
| 43. | 0.0129 | 0.0131 | 0.0146 | 0.0137 | 0.0145 | 0.0134 | 0.0129 | 0.0129 |
| 44. | 0.0136 | 0.0174 | 0.0136 | 0.0136 | 0.0133 | 0.0167 | 0.0133 | 0.0173 |
| 45. | 0.0130 | 0.0131 | 0.0132 | 0.0132 | 0.0132 | 0.0133 | 0.0131 | 0.0134 |
| 46. | 0.0131 | 0.0131 | 0.0130 | 0.0132 | 0.0131 | 0.0132 | 0.0131 | 0.0131 |
| 47. | 0.0132 | 0.0132 | 0.0132 | 0.0138 | 0.0131 | 0.0128 | 0.0131 | 0.0129 |
| 48. | 0.0133 | 0.0130 | 0.0133 | 0.0133 | 0.0132 | 0.0137 | 0.0131 | 0.0131 |
| 49. | 0.0131 | 0.0132 | 0.0132 | 0.0132 | 0.0131 | 0.0132 | 0.0132 | 0.0131 |
| 50. | 0.0131 | 0.0129 | 0.0132 | 0.0130 | 0.0130 | 0.0131 | 0.0131 | 0.0130 |
| 51. | 0.0137 | 0.0132 | 0.0129 | 0.0132 | 0.0130 | 0.0132 | 0.0132 | 0.0129 |
| 52. | 0.0132 | 0.0132 | 0.0133 | 0.0134 | 0.0132 | 0.0132 | 0.0132 | 0.0133 |
| 53. | 0.0128 | 0.0126 | 0.0132 | 0.0131 | 0.0132 | 0.0133 | 0.0128 | 0.0131 |
| 54. | 0.0130 | 0.0156 | 0.0162 | 0.0131 | 0.0133 | 0.0134 | 0.0130 | 0.01342 |
| 55. | 0.0131 | 0.0131 | 0.0132 | 0.0171 | 0.0130 | 0.0174 | 0.0130 | 0.0131 |
| 56. | 0.0130 | 0.0137 | 0.0179 | 0.0181 | 0.0178 | 0.0129 | 0.0185 | 0.0171 |
| 57. | 0.0133 | 0.0134 | 0.0133 | 0.0136 | 0.0135 | 0.0133 | 0.0136 | 0.0135 |
| 58. | 0.0131 | 0.0132 | 0.0132 | 0.0132 | 0.0131 | 0.0132 | 0.0132 | 0.0131 |
| 59. | 0.0133 | 0.0135 | 0.0128 | 0.0131 | 0.0132 | 0.0130 | 0.0130 | 0.0130 |
| 60. | 0.0134 | 0.0137 | 0.0131 | 0.0131 | 0.0133 | 0.0132 | 0.0132 | 0.0134 |
| 61. | 0.0129 | 0.0131 | 0.0131 | 0.0164 | 0.0131 | 0.0132 | 0.0134 | 0.0132 |
| 62. | 0.0191 | 0.0132 | 0.0187 | 0.0132 | 0.0138 | 0.0132 | 0.0173 | 0.0129 |
| 63. | 0.0130 | 0.0130 | 0.0132 | 0.0132 | 0.0129 | 0.0130 | 0.0129 | 0.0129 |
| 64. | 0.0131 | 0.0130 | 0.0131 | 0.0167 | 0.0131 | 0.0145 | 0.0132 | 0.0131 |
| 65. | 0.0128 | 0.0139 | 0.0131 | 0.0133 | 0.0132 | 0.0133 | 0.0133 | 0.0134 |
| 66. | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0132 | 0.0134 | 0.0134 | 0.0133 |
| 67. | 0.0132 | 0.0131 | 0.0132 | 0.0130 | 0.0131 | 0.0132 | 0.0138 | 0.0132 |
| 68. | 0.0129 | 0.0131 | 0.0132 | 0.0129 | 0.0129 | 0.0132 | 0.0131 | 0.0131 |
| 69. | 0.0132 | 0.0133 | 0.0134 | 0.0133 | 0.0132 | 0.0131 | 0.0132 | 0.0133 |
| 70. | 0.0185 | 0.0132 | 0.0134 | 0.0131 | 0.0132 | 0.0128 | 0.0130 | 0.0131 |
| 71. | 0.0146 | 0.0132 | 0.0127 | 0.0128 | 0.0128 | 0.0131 | 0.0127 | 0.0137 |

**For Oily**

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 0.0129 | 0.0153 | 0.0129 | 0.0158 | 0.0141 | 0.0138 | 0.0128 | 0.0153 |
| 2. | 0.0130 | 0.0122 | 0.0127 | 0.0138 | 0.0131 | 0.0131 | 0.0144 | 0.0139 |
| 3. | 0.0131 | 0.0132 | 0.0130 | 0.0129 | 0.0130 | 0.0118 | 0.0128 | 0.0129 |
| 4. | 0.0131 | 0.0134 | 0.0131 | 0.0132 | 0.0130 | 0.0131 | 0.0131 | 0.0131 |
| 5. | 0.0131 | 0.0130 | 0.0145 | 0.0132 | 0.0132 | 0.0130 | 0.0132 | 0.0132 |
| 6. | 0.0130 | 0.0133 | 0.0130 | 0.0130 | 0.0129 | 0.0129 | 0.0131 | 0.0131 |
| 7. | 0.0122 | 0.0132 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 |
| 8. | 0.0131 | 0.0133 | 0.0130 | 0.0130 | 0.0129 | 0.0130 | 0.0130 | 0.0130 |
| 9. | 0.0129 | 0.0130 | 0.0129 | 0.0128 | 0.0127 | 0.0126 | 0.0125 | 0.0129 |
| 10. | 0.0128 | 0.0130 | 0.0129 | 0.0130 | 0.0122 | 0.0124 | 0.0129 | 0.0129 |
| 11. | 0.0130 | 0.0130 | 0.0129 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0139 |
| 12. | 0.0129 | 0.0131 | 0.0131 | 0.0130 | 0.0131 | 0.0130 | 0.0133 | 0.0129 |
| 13. | 0.0127 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0131 |
| 14. | 0.0129 | 0.0130 | 0.0135 | 0.0129 | 0.0130 | 0.0129 | 0.0130 | 0.0129 |
| 15. | 0.0104 | 0.0104 | 0.0111 | 0.0110 | 0.0108 | 0.0135 | 0.0125 | 0.0130 |
| 16. | 0.0130 | 0.0129 | 0.0131 | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0130 |
| 17. | 0.0115 | 0.0151 | 0.0149 | 0.0135 | 0.0126 | 0.0129 | 0.0123 | 0.0121 |
| 18. | 0.0130 | 0.0124 | 0.0130 | 0.0133 | 0.0128 | 0.0129 | 0.0126 | 0.0130 |
| 19. | 0.0121 | 0.0132 | 0.0129 | 0.0107 | 0.0151 | 0.0107 | 0.0139 | 0.0123 |
| 20. | 0.0102 | 0.0114 | 0.0103 | 0.0130 | 0.0122 | 0.0118 | 0.0110 | 0.0101 |
| 21. | 0.0103 | 0.0108 | 0.0102 | 0.0123 | 0.0130 | 0.0125 | 0.0121 | 0.0131 |
| 22. | 0.0107 | 0.0106 | 0.0109 | 0.0132 | 0.0117 | 0.0137 | 0.0146 | 0.0128 |
| 23. | 0.0133 | 0.0129 | 0.0131 | 0.0130 | 0.0130 | 0.0130 | 0.0129 | 0.0132 |
| 24. | 0.0128 | 0.0124 | 0.0111 | 0.0121 | 0.0128 | 0.0127 | 0.0120 | 0.0120 |
| 25. | 0.0124 | 0.0132 | 0.0123 | 0.0124 | 0.0120 | 0.0135 | 0.0134 | 0.0130 |
| 26. | 0.0126 | 0.0132 | 0.0129 | 0.0130 | 0.0131 | 0.0127 | 0.0131 | 0.0130 |
| 27. | 0.0130 | 0.0130 | 0.0130 | 0.0138 | 0.0130 | 0.0130 | 0.0130 | 0.0132 |
| 28. | 0.0127 | 0.0130 | 0.0129 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0129 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 29. | 0.0151 | 0.0137 | 0.0138 | 0.0131 | 0.0147 | 0.0137 | 0.0127 | 0.0128 |
| 30. | 0.0111 | 0.0112 | 0.0114 | 0.0115 | 0.0117 | 0.0118 | 0.0112 | 0.0121 |
| 31. | 0.0142 | 0.0127 | 0.0130 | 0.0129 | 0.0127 | 0.0128 | 0.0135 | 0.0137 |
| 32. | 0.0153 | 0.0129 | 0.0155 | 0.0163 | 0.0152 | 0.0142 | 0.0149 | 0.0134 |
| 33. | 0.0144 | 0.0126 | 0.0130 | 0.0133 | 0.0143 | 0.0137 | 0.0127 | 0.0131 |
| 34. | 0.0134 | 0.0125 | 0.0126 | 0.0125 | 0.0131 | 0.0132 | 0.0130 | 0.0130 |
| 35. | 0.0152 | 0.0150 | 0.0139 | 0.0126 | 0.0141 | 0.0159 | 0.0142 | 0.0130 |
| 36. | 0.0131 | 0.0130 | 0.0131 | 0.0107 | 0.0114 | 0.0123 | 0.0130 | 0.0129 |
| 37. | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0130 | 0.0140 | 0.0156 | 0.0151 |
| 38. | 0.0133 | 0.0131 | 0.0129 | 0.0117 | 0.0119 | 0.0141 | 0.0136 | 0.0131 |
| 39. | 0.0131 | 0.0133 | 0.0131 | 0.0129 | 0.0118 | 0.0124 | 0.0128 | 0.0129 |
| 40. | 0.0129 | 0.0132 | 0.0121 | 0.0135 | 0.0137 | 0.0118 | 0.0127 | 0.0130 |
| 41. | 0.0118 | 0.0128 | 0.0132 | 0.0127 | 0.0123 | 0.0121 | 0.0114 | 0.0126 |
| 42. | 0.0121 | 0.0129 | 0.0140 | 0.0130 | 0.0133 | 0.0131 | 0.0130 | 0.0114 |
| 43. | 0.0111 | 0.0121 | 0.0128 | 0.0116 | 0.0138 | 0.0110 | 0.0123 | 0.0118 |
| 44. | 0.0130 | 0.0136 | 0.0125 | 0.0130 | 0.0138 | 0.0130 | 0.0130 | 0.0129 |
| 45. | 0.0139 | 0.0146 | 0.0142 | 0.0149 | 0.0130 | 0.0134 | 0.0143 | 0.0131 |
| 46. | 0.0130 | 0.0131 | 0.0132 | 0.0130 | 0.0128 | 0.0130 | 0.0131 | 0.0127 |
| 47. | 0.0149 | 0.0129 | 0.0132 | 0.0130 | 0.0126 | 0.0129 | 0.0133 | 0.0142 |
| 48. | 0.0121 | 0.0130 | 0.0127 | 0.0122 | 0.0129 | 0.0130 | 0.0128 | 0.0134 |
| 49. | 0.0123 | 0.0122 | 0.0131 | 0.0135 | 0.0127 | 0.0135 | 0.0129 | 0.0120 |
| 50. | 0.0131 | 0.0129 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0131 |
| 51. | 0.0128 | 0.0130 | 0.0127 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 |
| 52. | 0.0131 | 0.0130 | 0.0130 | 0.0141 | 0.0147 | 0.0144 | 0.0131 | 0.0130 |
| 53. | 0.0127 | 0.0128 | 0.0124 | 0.0130 | 0.0127 | 0.0128 | 0.0129 | 0.0132 |
| 54. | 0.0133 | 0.0138 | 0.0126 | 0.0132 | 0.0124 | 0.0117 | 0.0127 | 0.0131 |
| 55. | 0.0122 | 0.0128 | 0.0125 | 0.0125 | 0.0114 | 0.0119 | 0.0129 | 0.0129 |
| 56. | 0.0122 | 0.0132 | 0.0123 | 0.0134 | 0.0145 | 0.0130 | 0.0136 | 0.0156 |
| 57. | 0.0135 | 0.0130 | 0.0132 | 0.0132 | 0.0132 | 0.0147 | 0.0144 | 0.0132 |
| 58. | 0.0123 | 0.0122 | 0.0131 | 0.0135 | 0.0127 | 0.0135 | 0.0129 | 0.0120 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 59. | 0.0133 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 |
| 60. | 0.0129 | 0.0132 | 0.0131 | 0.0129 | 0.0130 | 0.0129 | 0.0130 | 0.0130 |
| 61. | 0.0129 | 0.0128 | 0.0133 | 0.0130 | 0.0131 | 0.0142 | 0.0130 | 0.0130 |
| 62. | 0.0149 | 0.0137 | 0.0130 | 0.0131 | 0.0132 | 0.0130 | 0.0134 | 0.0129 |
| 63. | 0.0155 | 0.0131 | 0.0130 | 0.0135 | 0.0129 | 0.0126 | 0.0122 | 0.0134 |
| 64. | 0.0123 | 0.0132 | 0.0131 | 0.0130 | 0.0116 | 0.0130 | 0.0131 | 0.0130 |
| 65. | 0.0130 | 0.0129 | 0.0134 | 0.0130 | 0.0127 | 0.0124 | 0.0129 | 0.0125 |
| 66. | 0.0130 | 0.0130 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0129 | 0.0129 |
| 67. | 0.0123 | 0.0129 | 0.0130 | 0.0131 | 0.0132 | 0.0112 | 0.0125 | 0.0134 |
| 68. | 0.0131 | 0.0130 | 0.0138 | 0.0130 | 0.0131 | 0.0130 | 0.0130 | 0.0130 |
| 69. | 0.0130 | 0.0129 | 0.0131 | 0.0130 | 0.0134 | 0.0129 | 0.0130 | 0.0130 |
| 70. | 0.0137 | 0.0136 | 0.0115 | 0.0121 | 0.0122 | 0.0130 | 0.0130 | 0.0127 |
| 71. | 0.0130 | 0.0129 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0130 | 0.0129 |

# Appendix G

## The RVTR for Dry , Oily, and Neutral Fingerprint Image

## For Dry:

| FINGER / Person | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 6.7557e-005 | 6.9074e-005 | 7.1427e-005 | 7.6393e-005 | 7.3589e-005 | 6.7169e-005 | 7.1224e-005 | 7.1282e-005 |
| 2. | 6.6096e-005 | 7.0580e-005 | 7.0274e-005 | 6.9991e-005 | 7.0595e-005 | 6.9732e-005 | 7.0501e-005 | 6.9274e-005 |
| 3. | 7.2351e-005 | 6.7538e-005 | 7.2062e-005 | 6.7047e-005 | 6.5351e-005 | 6.6496e-005 | 7.2768e-005 | 6.9722e-005 |
| 4. | 6.6639e-005 | 6.5760e-005 | 6.7272e-005 | 6.7639e-005 | 6.5061e-005 | 6.9698e-005 | 6.7198e-005 | 7.2071e-005 |
| 5. | 6.4641e-005 | 6.6624e-005 | 6.4758e-005 | 6.8665e-005 | 6.5816e-005 | 6.6588e-005 | 7.1107e-005 | 6.5322e-005 |
| 6. | 6.7217e-005 | 6.7685e-005 | 6.5558e-005 | 6.6158e-005 | 6.6808e-005 | 6.8684e-005 | 6.8603e-005 | 6.6808e-005 |
| 7. | 6.8431e-005 | 6.6019e-005 | 7.0428e-005 | 7.0746e-005 | 6.7458e-005 | 6.5847e-005 | 6.8724e-005 | 7.0724e-005 |
| 8. | 6.5428e-005 | 6.7265e-005 | 6.4173e-005 | 6.7124e-005 | 6.3950e-005 | 6.7965e-005 | 6.3070e-005 | 6.4673e-005 |
| 9. | 6.2755e-005 | 6.3926e-005 | 6.2754e-005 | 6.1443e-005 | 6.0622e-005 | 6.2608e-005 | 6.7797e-005 | 6.5276e-005 |
| 10. | 7.2156e-005 | 7.1226e-005 | 7.4735e-005 | 7.3667e-005 | 7.2057e-005 | 7.4825e-005 | 7.3282e-005 | 7.1493e-005 |
| 11. | 6.4636e-005 | 6.5400e-005 | 6.1447e-005 | 6.1805e-005 | 6.1808e-005 | 6.0367e-005 | 5.9686e-005 | 6.3777e-005 |
| 12. | 5.9396e-005 | 5.9823e-005 | 6.0736e-005 | 6.3255e-005 | 6.0329e-005 | 6.2011e-005 | 6.0812e-005 | 6.1229e-005 |
| 13. | 6.1436e-005 | 6.3353e-005 | 6.4445e-005 | 6.2834e-005 | 6.3667e-005 | 6.2906e-005 | 6.0277e-005 | 6.4344e-005 |
| 14. | 6.7841e-005 | 6.3055e-005 | 6.5662e-005 | 6.3961e-005 | 6.1443e-005 | 6.5306e-005 | 6.4931e-005 | 6.3895e-005 |
| 15. | 6.8155e-005 | 6.8420e-005 | 6.9555e-005 | 6.7186e-005 | 6.8567e-005 | 6.9338e-005 | 6.4410e-005 | 6.7673e-005 |
| 16. | 6.1143e-005 | 6.5210e-005 | 6.6227e-005 | 6.6154e-005 | 6.8071e-005 | 6.6047e-005 | 7.3056e-005 | 7.0756e-005 |
| 17. | 7.0863e-005 | 7.4834e-005 | 7.3417e-005 | 6.6623e-005 | 6.9624e-005 | 7.2894e-005 | 7.2639e-005 | 6.8400e-005 |
| 18. | 6.7569e-005 | 6.4868e-005 | 6.9766e-005 | 6.9352e-005 | 6.4292e-005 | 6.1036e-005 | 6.9348e-005 | 6.9931e-005 |
| 19. | 7.1828e-005 | 7.0986e-005 | 7.1534e-005 | 7.1916e-005 | 7.3568e-005 | 7.4425e-005 | 7.2997e-005 | 6.8629e-005 |
| 20. | 7.3475e-005 | 7.3475e-005 | 7.0314e-005 | 6.8189e-005 | 7.2177e-005 | 7.1937e-005 | 7.2687e-005 | 7.4595e-005 |
| 21. | 6.4367e-005 | 6.7008e-005 | 6.6667e-005 | 6.6489e-005 | 6.2617e-005 | 6.5449e-005 | 6.3243e-005 | 6.9932e-005 |
| 22. | 7.4322e-005 | 7.4278e-005 | 7.5352e-005 | 7.4601e-005 | 7.1108e-005 | 7.0907e-005 | 7.3319e-005 | 7.1931e-005 |
| 23. | 5.9744e-005 | 6.3776e-005 | 6.1024e-005 | 6.2936e-005 | 6.1383e-005 | 6.1790e-005 | 5.9693e-005 | 6.3812e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 24. | 6.5392e-005 | 6.7352e-005 | 7.2311e-005 | 7.0793e-005 | 6.9560e-005 | 7.3167e-005 | 7.2789e-005 | 7.0186e-005 |
| 25. | 6.7899e-005 | 6.8649e-005 | 6.8413e-005 | 7.4574e-005 | 6.8235e-005 | 6.8345e-005 | 7.1091e-005 | 6.8876e-005 |
| 26. | 6.8906e-005 | 7.1437e-005 | 7.2523e-005 | 7.3691e-005 | 7.0162e-005 | 6.9366e-005 | 7.0935e-005 | 6.9380e-005 |
| 27. | 6.9361e-005 | 6.7164e-005 | 6.9109e-005 | 6.9850e-005 | 6.7165e-005 | 7.1149e-005 | 6.8590e-005 | 6.7581e-005 |
| 28. | 6.3946e-005 | 6.0723e-005 | 6.1184e-005 | 6.3992e-005 | 5.9940e-005 | 6.4677e-005 | 6.0865e-005 | 6.4366e-005 |
| 29. | 6.6147e-005 | 7.3254e-005 | 7.2546e-005 | 6.9414e-005 | 6.9142e-005 | 6.8047e-005 | 6.6258e-005 | 7.0470e-005 |
| 30. | 7.2742e-005 | 7.2311e-005 | 6.7130e-005 | 6.4961e-005 | 6.5576e-005 | 7.2585e-005 | 6.9894e-005 | 6.7164e-005 |
| 31. | 6.4365e-005 | 7.0113e-005 | 6.5716e-005 | 7.1919e-005 | 6.9147e-005 | 6.6256e-005 | 6.5673e-005 | 6.8163e-005 |
| 32. | 6.5858e-005 | 7.2374e-005 | 6.6138e-005 | 7.4230e-005 | 7.1827e-005 | 6.6659e-005 | 6.4332e-005 | 6.9763e-005 |
| 33. | 6.5952e-005 | 7.0584e-005 | 7.1785e-005 | 6.7550e-005 | 6.5672e-005 | 6.7170e-005 | 6.8248e-005 | 6.6777e-005 |
| 34. | 6.7085e-005 | 6.5588e-005 | 6.5731e-005 | 6.5213e-005 | 6.3949e-005 | 6.8974e-005 | 6.5852e-005 | 6.5386e-005 |
| 35. | 6.6842e-005 | 6.8447e-005 | 6.1829e-005 | 6.5204e-005 | 6.9027e-005 | 6.1048e-005 | 6.9801e-005 | 5.9278e-005 |
| 36. | 6.7351e-005 | 6.9013e-005 | 7.0798e-005 | 6.6844e-005 | 7.0632e-005 | 6.8582e-005 | 6.8814e-005 | 7.0864e-005 |
| 37. | 6.4717e-005 | 6.2237e-005 | 6.0936e-005 | 6.1574e-005 | 6.0554e-005 | 6.4722e-005 | 6.2644e-005 | 6.3194e-005 |
| 38. | 7.2621e-005 | 7.1206e-005 | 7.3118e-005 | 6.9840e-005 | 7.4388e-005 | 6.8356e-005 | 6.8524e-005 | 7.0205e-005 |
| 39. | 5.8187e-005 | 6.9866e-005 | 6.4192e-005 | 6.4805e-005 | 6.6717e-005 | 7.1905e-005 | 6.6315e-005 | 6.6964e-005 |
| 40. | 6.8396e-005 | 6.5635e-005 | 6.5908e-005 | 6.6310e-005 | 6.6199e-005 | 6.7801e-005 | 6.4002e-005 | 6.5217e-005 |
| 41. | 6.4785e-005 | 6.5743e-005 | 6.4755e-005 | 6.4299e-005 | 6.6473e-005 | 6.2879e-005 | 6.1832e-005 | 6.3829e-005 |
| 42. | 7.4243e-005 | 6.8514e-005 | 6.9911e-005 | 7.0831e-005 | 7.1869e-005 | 7.0683e-005 | 6.9418e-005 | 6.9364e-005 |
| 43. | 7.2874e-005 | 7.1028e-005 | 7.0561e-005 | 6.5799e-005 | 7.1136e-005 | 7.1566e-005 | 7.1791e-005 | 6.7767e-005 |
| 44. | 6.2672e-005 | 7.2361e-005 | 6.6889e-005 | 6.3501e-005 | 6.3999e-005 | 6.8779e-005 | 6.4627e-005 | 6.5693e-005 |
| 45. | 7.1925e-005 | 6.4918e-005 | 6.6256e-005 | 6.4024e-005 | 6.7033e-005 | 6.4992e-005 | 6.2497e-005 | 6.8619e-005 |
| 46. | 6.2222e-005 | 6.0356e-005 | 6.2070e-005 | 6.1940e-005 | 6.1343e-005 | 6.2823e-005 | 6.6197e-005 | 6.2483e-005 |
| 47. | 6.5499e-005 | 6.2534e-005 | 6.3004e-005 | 6.2971e-005 | 6.0990e-005 | 6.2886e-005 | 6.4578e-005 | 6.4903e-005 |
| 48. | 6.4645e-005 | 6.4212e-005 | 6.3443e-005 | 6.6045e-005 | 6.2156e-005 | 6.3142e-005 | 6.5935e-005 | 6.4247e-005 |
| 49. | 6.5157e-005 | 6.5067e-005 | 6.3491e-005 | 6.2269e-005 | 6.2781e-005 | 6.1849e-005 | 6.6367e-005 | 6.2521e-005 |
| 50. | 6.3806e-005 | 6.5444e-005 | 6.5653e-005 | 6.1508e-005 | 5.9142e-005 | 6.2071e-005 | 5.9054e-005 | 5.8974e-005 |
| 51. | 6.0932e-005 | 6.0707e-005 | 6.0965e-005 | 5.8766e-005 | 5.8764e-005 | 6.3558e-005 | 5.9257e-005 | 6.3542e-005 |
| 52. | 6.3243e-005 | 6.2466e-005 | 6.1665e-005 | 6.1471e-005 | 6.1037e-005 | 6.2176e-005 | 6.1646e-005 | 6.2993e-005 |
| 53. | 7.2148e-005 | 6.8365e-005 | 7.3485e-005 | 7.3923e-005 | 6.6028e-005 | 6.7564e-005 | 7.0424e-005 | 7.0992e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 54. | 6.4991e-005 | 6.3369e-005 | 6.1693e-005 | 6.2201e-005 | 6.1303e-005 | 6.4987e-005 | 6.1576e-005 | 6.5628e-005 |
| 55. | 6.4627e-005 | 6.4093e-005 | 6.4361e-005 | 6.3487e-005 | 7.0017e-005 | 6.5560e-005 | 6.2279e-005 | 6.6416e-005 |
| 56. | 6.6927e-005 | 7.1768e-005 | 6.6674e-005 | 6.5879e-005 | 6.2768e-005 | 6.7504e-005 | 6.3665e-005 | 6.8770e-005 |
| 57. | 6.5237e-005 | 6.7950e-005 | 6.7438e-005 | 7.1762e-005 | 6.5350e-005 | 6.7502e-005 | 6.5742e-005 | 6.6055e-005 |
| 58. | 6.5157e-005 | 6.5067e-005 | 6.3491e-005 | 6.2269e-005 | 6.2781e-005 | 6.1849e-005 | 6.6367e-005 | 6.2521e-005 |
| 59. | 6.5833e-005 | 6.7695e-005 | 6.4951e-005 | 6.5434e-005 | 6.5844e-005 | 6.6458e-005 | 6.6929e-005 | 6.6179e-005 |
| 60. | 6.8145e-005 | 6.8675e-005 | 7.0873e-005 | 6.8305e-005 | 6.7456e-005 | 6.6950e-005 | 6.6039e-005 | 6.6614e-005 |
| 61. | 6.6462e-005 | 5.9314e-005 | 6.1057e-005 | 6.5367e-005 | 5.9537e-005 | 6.4033e-005 | 6.4839e-005 | 7.0197e-005 |
| 62. | 6.3859e-005 | 6.0313e-005 | 5.9612e-005 | 5.8969e-005 | 6.0125e-005 | 6.0692e-005 | 6.0536e-005 | 6.0915e-005 |
| 63. | 6.3012e-005 | 6.5691e-005 | 6.2958e-005 | 6.3316e-005 | 6.3647e-005 | 6.2518e-005 | 6.4482e-005 | 6.1440e-005 |
| 64. | 6.7011e-005 | 6.8150e-005 | 6.8198e-005 | 6.8305e-005 | 6.7824e-005 | 6.8676e-005 | 6.8476e-005 | 6.6794e-005 |
| 65. | 6.7416e-005 | 6.6657e-005 | 6.6500e-005 | 6.5659e-005 | 6.7508e-005 | 6.7752e-005 | 6.7477e-005 | 7.0586e-005 |
| 66. | 6.2197e-005 | 6.0476e-005 | 6.5580e-005 | 6.2909e-005 | 6.3980e-005 | 6.0711e-005 | 6.0031e-005 | 6.3323e-005 |
| 67. | 5.9696e-005 | 5.9334e-005 | 6.1504e-005 | 6.3107e-005 | 6.0592e-005 | 6.1429e-005 | 6.1118e-005 | 6.1284e-005 |
| 68. | 6.2910e-005 | 6.4151e-005 | 6.7294e-005 | 6.3491e-005 | 6.4111e-005 | 6.5417e-005 | 6.2736e-005 | 6.0064e-005 |
| 69. | 6.8357e-005 | 6.7239e-005 | 7.0692e-005 | 6.3011e-005 | 6.6284e-005 | 6.5563e-005 | 6.6417e-005 | 6.4701e-005 |
| 70. | 6.4623e-005 | 6.6231e-005 | 7.0210e-005 | 6.3523e-005 | 6.1732e-005 | 6.7106e-005 | 7.0411e-005 | 6.3845e-005 |
| 71. | 6.1290e-005 | 6.2688e-005 | 6.1584e-005 | 6.0716e-005 | 6.1188e-005 | 5.8695e-005 | 5.9094e-005 | 6.0101e-005 |

## For Neutral :

| FINGER / Person | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
|---|---|---|---|---|---|---|---|---|
| 1. | 6.9037e-005 | 6.9613e-005 | 6.9779e-005 | 7.1505e-005 | 6.8312e-005 | 6.8731e-005 | 6.9732e-005 | 7.0054e-005 |
| 2. | 6.7974e-005 | 6.3096e-005 | 6.6922e-005 | 6.9151e-005 | 6.6895e-005 | 6.7987e-005 | 6.6807e-005 | 6.7285e-005 |
| 3. | 6.6985e-005 | 5.9779e-005 | 6.3770e-005 | 6.6843e-005 | 6.0812e-005 | 5.9939e-005 | 6.4536e-005 | 6.3459e-005 |
| 4. | 6.4860e-005 | 6.7721e-005 | 6.2999e-005 | 6.2821e-005 | 6.3998e-005 | 6.3225e-005 | 6.4744e-005 | 6.3695e-005 |
| 5. | 6.6191e-005 | 6.5079e-005 | 6.3057e-005 | 7.0652e-005 | 6.6317e-005 | 6.2335e-005 | 6.3038e-005 | 7.0346e-005 |
| 6. | 6.8606e-005 | 6.5672e-005 | 6.8709e-005 | 6.4341e-005 | 6.5472e-005 | 6.8140e-005 | 6.6426e-005 | 6.6697e-005 |
| 7. | 7.3957e-005 | 7.1498e-005 | 7.2182e-005 | 7.0347e-005 | 6.5796e-005 | 6.9691e-005 | 7.0394e-005 | 6.5338e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 8. | 6.7923e-005 | 6.2572e-005 | 6.4817e-005 | 6.2264e-005 | 6.2734e-005 | 6.2546e-005 | 6.4443e-005 | 6.0645e-005 |
| 9. | 5.7197e-005 | 5.7194e-005 | 5.6957e-005 | 5.6906e-005 | 5.7039e-005 | 5.7149e-005 | 5.7628e-005 | 5.6981e-005 |
| 10. | 7.2485e-005 | 7.1425e-005 | 7.2449e-005 | 7.3565e-005 | 7.3135e-005 | 7.3098e-005 | 7.1771e-005 | 7.2919e-005 |
| 11. | 7.0703e-005 | 6.3531e-005 | 6.4254e-005 | 6.6870e-005 | 6.1439e-005 | 6.2656e-005 | 6.1279e-005 | 6.3679e-005 |
| 12. | 6.4494e-005 | 6.2804e-005 | 5.9651e-005 | 6.0465e-005 | 6.1821e-005 | 6.1686e-005 | 5.9498e-005 | 5.8230e-005 |
| 13. | 5.9051e-005 | 5.9800e-005 | 6.0544e-005 | 5.9761e-005 | 6.1043e-005 | 6.2609e-005 | 5.8358e-005 | 6.1991e-005 |
| 14. | 6.8390e-005 | 6.2293e-005 | 5.9511e-005 | 5.9032e-005 | 5.8306e-005 | 5.8856e-005 | 5.8578e-005 | 5.8658e-005 |
| 15. | 7.1030e-005 | 6.7342e-005 | 6.8032e-005 | 6.8465e-005 | 6.4230e-005 | 6.6969e-005 | 6.0966e-005 | 6.6653e-005 |
| 16. | 6.9808e-005 | 6.2900e-005 | 6.7400e-005 | 6.4108e-005 | 6.2099e-005 | 6.3062e-005 | 6.2427e-005 | 6.1203e-005 |
| 17. | 7.1492e-005 | 7.3784e-005 | 6.9280e-005 | 6.7444e-005 | 6.7626e-005 | 6.5536e-005 | 6.8969e-005 | 6.7051e-005 |
| 18. | 6.2305e-005 | 6.4313e-005 | 6.6668e-005 | 6.3304e-005 | 7.0897e-005 | 6.0996e-005 | 6.5376e-005 | 6.6423e-005 |
| 19. | 6.2617e-005 | 6.9489e-005 | 6.5787e-005 | 6.7875e-005 | 6.9308e-005 | 7.1455e-005 | 6.6778e-005 | 7.0626e-005 |
| 20. | 7.7842e-005 | 7.1989e-005 | 7.3784e-005 | 7.3393e-005 | 6.9860e-005 | 7.1476e-005 | 7.1592e-005 | 7.2837e-005 |
| 21. | 6.6631e-005 | 6.5886e-005 | 6.6895e-005 | 7.0835e-005 | 6.9196e-005 | 6.9541e-005 | 6.6637e-005 | 6.5512e-005 |
| 22. | 7.5196e-005 | 7.0723e-005 | 7.3297e-005 | 7.1395e-005 | 6.7828e-005 | 7.1688e-005 | 6.7068e-005 | 6.8255e-005 |
| 23. | 5.8975e-005 | 5.9089e-005 | 5.8817e-005 | 5.8070e-005 | 5.8141e-005 | 5.8257e-005 | 6.0221e-005 | 6.2906e-005 |
| 24. | 7.2007e-005 | 6.4899e-005 | 6.9139e-005 | 6.8549e-005 | 6.8970e-005 | 7.2787e-005 | 7.5253e-005 | 6.9438e-005 |
| 25. | 7.0194e-005 | 6.8216e-005 | 6.7841e-005 | 6.7564e-005 | 6.7651e-005 | 7.0230e-005 | 6.7968e-005 | 6.8357e-005 |
| 26. | 6.8616e-005 | 6.3670e-005 | 6.3508e-005 | 6.8052e-005 | 6.7852e-005 | 7.0729e-005 | 6.8942e-005 | 6.7778e-005 |
| 27. | 5.9221e-005 | 5.8077e-005 | 6.0157e-005 | 5.7955e-005 | 5.8766e-005 | 5.8244e-005 | 5.8565e-005 | 5.6558e-005 |
| 28. | 5.9449e-005 | 5.6608e-005 | 6.1207e-005 | 5.6381e-005 | 5.6429e-005 | 5.5982e-005 | 5.5712e-005 | 5.6010e-005 |
| 29. | 6.8495e-005 | 6.4577e-005 | 6.6619e-005 | 6.4356e-005 | 6.5113e-005 | 6.5267e-005 | 6.7803e-005 | 6.8022e-005 |
| 30. | 6.9916e-005 | 6.6502e-005 | 6.8221e-005 | 6.2153e-005 | 6.1689e-005 | 6.1048e-005 | 6.0354e-005 | 6.9053e-005 |
| 31. | 6.7301e-005 | 6.8130e-005 | 6.9310e-005 | 6.8780e-005 | 6.7491e-005 | 6.8067e-005 | 6.7301e-005 | 6.6592e-005 |
| 32. | 6.4437e-005 | 6.1822e-005 | 6.2377e-005 | 6.2888e-005 | 6.2526e-005 | 6.2250e-005 | 6.2750e-005 | 6.9177e-005 |
| 33. | 7.0886e-005 | 6.7604e-005 | 7.2679e-005 | 6.7452e-005 | 7.1889e-005 | 6.9943e-005 | 6.7724e-005 | 6.5559e-005 |
| 34. | 6.2541e-005 | 6.5808e-005 | 6.7119e-005 | 6.5905e-005 | 6.6566e-005 | 6.6746e-005 | 6.4958e-005 | 6.6275e-005 |
| 35. | 6.4833e-005 | 5.9655e-005 | 5.9845e-005 | 5.8743e-005 | 5.9211e-005 | 6.3467e-005 | 5.8236e-005 | 5.8665e-005 |
| 36. | 6.6450e-005 | 6.9626e-005 | 7.2152e-005 | 7.1033e-005 | 6.9001e-005 | 6.6484e-005 | 6.3850e-005 | 6.6998e-005 |
| 37. | 5.9282e-005 | 5.7917e-005 | 6.2287e-005 | 6.0481e-005 | 5.7031e-005 | 5.8105e-005 | 5.6706e-005 | 5.7595e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 38. | 6.7143e-005 | 6.5065e-005 | 6.8372e-005 | 6.5881e-005 | 6.7407e-005 | 6.4884e-005 | 6.9380e-005 | 6.9659e-005 |
| 39. | 6.5785e-005 | 6.4622e-005 | 6.0922e-005 | 6.0013e-005 | 6.7886e-005 | 5.9705e-005 | 5.7444e-005 | 5.6462e-005 |
| 40. | 6.0548e-005 | 6.6394e-005 | 6.4077e-005 | 6.4119e-005 | 6.3342e-005 | 6.3349e-005 | 6.5470e-005 | 6.4855e-005 |
| 41. | 6.9923e-005 | 6.6722e-005 | 6.4263e-005 | 6.6314e-005 | 6.5099e-005 | 6.5159e-005 | 6.3393e-005 | 6.4023e-005 |
| 42. | 6.8791e-005 | 6.5102e-005 | 6.9524e-005 | 7.0513e-005 | 6.9380e-005 | 6.9690e-005 | 6.8339e-005 | 6.7824e-005 |
| 43. | 6.3934e-005 | 6.0026e-005 | 6.9836e-005 | 6.9157e-005 | 7.1261e-005 | 7.0419e-005 | 7.3388e-005 | 7.1316e-005 |
| 44. | 7.2216e-005 | 7.0609e-005 | 6.7746e-005 | 7.1726e-005 | 7.0614e-005 | 6.7431e-005 | 6.6757e-005 | 6.7813e-005 |
| 45. | 6.8681e-005 | 6.4511e-005 | 6.4487e-005 | 6.3349e-005 | 6.2208e-005 | 6.5128e-005 | 6.3931e-005 | 6.1558e-005 |
| 46. | 6.1719e-005 | 5.9320e-005 | 6.2077e-005 | 6.0464e-005 | 5.9910e-005 | 6.1173e-005 | 6.2845e-005 | 5.8802e-005 |
| 47. | 6.4056e-005 | 6.4147e-005 | 6.7727e-005 | 6.7750e-005 | 6.6840e-005 | 6.7797e-005 | 6.3351e-005 | 6.6107e-005 |
| 48. | 6.7569e-005 | 6.8604e-005 | 6.4555e-005 | 6.7320e-005 | 6.6100e-005 | 6.6998e-005 | 6.9080e-005 | 6.3515e-005 |
| 49. | 6.0171e-005 | 5.7915e-005 | 5.8900e-005 | 5.8025e-005 | 6.5038e-005 | 5.9848e-005 | 5.8293e-005 | 6.2030e-005 |
| 50. | 6.5044e-005 | 6.5614e-005 | 6.3458e-005 | 6.1670e-005 | 6.3281e-005 | 5.9998e-005 | 6.0220e-005 | 6.2615e-005 |
| 51. | 5.9821e-005 | 6.1189e-005 | 6.2361e-005 | 6.2480e-005 | 5.9523e-005 | 5.7845e-005 | 6.1694e-005 | 6.0770e-005 |
| 52. | 6.3491e-005 | 5.9944e-005 | 6.1685e-005 | 6.2535e-005 | 6.2980e-005 | 6.1874e-005 | 6.1949e-005 | 6.1135e-005 |
| 53. | 6.9449e-005 | 7.2520e-005 | 6.9261e-005 | 6.7347e-005 | 6.9863e-005 | 6.5454e-005 | 7.0881e-005 | 7.0677e-005 |
| 54. | 6.4317e-005 | 6.3836e-005 | 6.5016e-005 | 6.3498e-005 | 6.0791e-005 | 6.5949e-005 | 6.5256e-005 | 6.5479e-005 |
| 55. | 6.3712e-005 | 6.6927e-005 | 6.4091e-005 | 6.9573e-005 | 6.4735e-005 | 6.5784e-005 | 6.5102e-005 | 6.5933e-005 |
| 56. | 6.7375e-005 | 6.7990e-005 | 6.9944e-005 | 6.9951e-005 | 7.1087e-005 | 6.3958e-005 | 6.9049e-005 | 6.7299e-005 |
| 57. | 6.1283e-005 | 6.3580e-005 | 7.0876e-005 | 6.6316e-005 | 6.4919e-005 | 6.2185e-005 | 6.9288e-005 | 6.4370e-005 |
| 58. | 6.0171e-005 | 5.7915e-005 | 5.8900e-005 | 5.8025e-005 | 6.5038e-005 | 5.9848e-005 | 5.8293e-005 | 6.2030e-005 |
| 59. | 6.3904e-005 | 6.8932e-005 | 6.6474e-005 | 6.6690e-005 | 6.5008e-005 | 6.5621e-005 | 6.5600e-005 | 6.5128e-005 |
| 60. | 6.5838e-005 | 6.4598e-005 | 6.3043e-005 | 6.5332e-005 | 6.5660e-005 | 6.4679e-005 | 6.5068e-005 | 6.4853e-005 |
| 61. | 6.3722e-005 | 6.0861e-005 | 5.9967e-005 | 6.1817e-005 | 6.1006e-005 | 5.8377e-005 | 5.7703e-005 | 5.7520e-005 |
| 62. | 6.2845e-005 | 6.3059e-005 | 6.3167e-005 | 6.3898e-005 | 6.0095e-005 | 6.0903e-005 | 6.2045e-005 | 6.2933e-005 |
| 63. | 6.1456e-005 | 6.5429e-005 | 6.0311e-005 | 6.3534e-005 | 6.5229e-005 | 6.3718e-005 | 6.5714e-005 | 6.3410e-005 |
| 64. | 6.8513e-005 | 6.5015e-005 | 6.5245e-005 | 6.7160e-005 | 6.3214e-005 | 6.4779e-005 | 6.3924e-005 | 6.6335e-005 |
| 65. | 6.9259e-005 | 7.1272e-005 | 7.2190e-005 | 6.8642e-005 | 6.7938e-005 | 6.8219e-005 | 6.6451e-005 | 6.7789e-005 |
| 66. | 6.5394e-005 | 5.9595e-005 | 6.2259e-005 | 5.8873e-005 | 6.0026e-005 | 5.9568e-005 | 5.8544e-005 | 5.8169e-005 |
| 67. | 5.9546e-005 | 6.1028e-005 | 6.0501e-005 | 6.2440e-005 | 6.1806e-005 | 6.1362e-005 | 6.2613e-005 | 5.8617e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 68. | 6.7339e-005 | 6.1972e-005 | 6.4604e-005 | 6.5035e-005 | 7.0464e-005 | 6.4707e-005 | 5.9022e-005 | 6.0472e-005 |
| 69. | 6.8139e-005 | 6.7624e-005 | 6.4836e-005 | 6.2439e-005 | 6.5307e-005 | 6.2833e-005 | 6.5431e-005 | 6.5623e-005 |
| 70. | 6.8667e-005 | 6.3156e-005 | 6.6383e-005 | 6.3604e-005 | 6.8194e-005 | 6.6875e-005 | 6.5323e-005 | 6.4336e-005 |
| 71. | 6.5540e-005 | 6.4579e-005 | 6.4350e-005 | 6.3062e-005 | 6.5376e-005 | 6.5902e-005 | 6.5032e-005 | 6.5656e-005 |

## For Oily:

| FINGER / Person | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 6.2187e-005 | 6.6594e-005 | 6.2187e-005 | 6.2016e-005 | 6.3980e-005 | 6.5731e-005 | 6.2943e-005 | 6.3699e-005 |
| 2. | 6.3509e-005 | 6.4963e-005 | 6.3405e-005 | 6.2624e-005 | 6.2206e-005 | 6.1204e-005 | 6.4238e-005 | 6.2864e-005 |
| 3. | 5.9531e-005 | 6.2708e-005 | 6.0159e-005 | 6.1588e-005 | 5.8654e-005 | 6.4825e-005 | 6.4301e-005 | 6.0450e-005 |
| 4. | 6.1751e-005 | 6.2338e-005 | 6.4445e-005 | 6.0764e-005 | 6.1436e-005 | 6.1785e-005 | 6.1117e-005 | 6.0366e-005 |
| 5. | 6.1924e-005 | 6.2002e-005 | 6.2884e-005 | 6.1312e-005 | 6.3681e-005 | 5.9533e-005 | 6.2175e-005 | 5.9714e-005 |
| 6. | 5.9621e-005 | 6.3192e-005 | 6.2682e-005 | 6.2690e-005 | 6.1590e-005 | 6.0134e-005 | 6.4322e-005 | 6.6201e-005 |
| 7. | 6.5345e-005 | 6.1523e-005 | 5.8701e-005 | 5.8064e-005 | 5.8165e-005 | 5.7671e-005 | 5.8794e-005 | 5.9364e-005 |
| 8. | 6.3250e-005 | 6.2022e-005 | 6.1359e-005 | 6.1671e-005 | 6.5250e-005 | 6.2885e-005 | 6.5779e-005 | 6.6246e-005 |
| 9. | 5.8155e-005 | 5.7718e-005 | 5.8187e-005 | 5.8831e-005 | 6.0749e-005 | 6.0072e-005 | 5.9705e-005 | 5.9071e-005 |
| 10. | 6.5837e-005 | 6.1414e-005 | 6.1351e-005 | 5.8834e-005 | 6.3715e-005 | 6.3100e-005 | 6.1120e-005 | 6.0739e-005 |
| 11. | 5.6613e-005 | 5.7712e-005 | 5.8345e-005 | 6.1530e-005 | 5.9232e-005 | 6.1982e-005 | 6.3730e-005 | 6.0879e-005 |
| 12. | 5.6365e-005 | 5.8138e-005 | 5.6831e-005 | 5.6826e-005 | 5.6518e-005 | 5.6394e-005 | 5.9030e-005 | 5.7106e-005 |
| 13. | 5.9419e-005 | 5.7536e-005 | 5.6816e-005 | 5.6638e-005 | 5.6972e-005 | 5.7133e-005 | 5.7706e-005 | 5.7386e-005 |
| 14. | 6.3083e-005 | 6.1005e-005 | 6.0654e-005 | 6.1883e-005 | 6.0356e-005 | 6.0114e-005 | 6.0135e-005 | 6.0433e-005 |
| 15. | 7.4633e-005 | 7.3961e-005 | 7.2356e-005 | 7.0079e-005 | 7.1624e-005 | 6.8076e-005 | 6.4040e-005 | 6.0300e-005 |
| 16. | 5.9224e-005 | 6.1750e-005 | 6.2220e-005 | 5.9758e-005 | 5.9864e-005 | 6.0492e-005 | 6.3572e-005 | 5.8676e-005 |
| 17. | 6.6338e-005 | 6.4281e-005 | 6.1788e-005 | 6.1283e-005 | 6.4790e-005 | 6.0884e-005 | 6.4247e-005 | 6.4006e-005 |
| 18. | 6.0493e-005 | 6.1666e-005 | 5.9161e-005 | 5.9381e-005 | 5.8643e-005 | 5.8125e-005 | 6.0031e-005 | 5.9303e-005 |
| 19. | 7.3469e-005 | 6.5654e-005 | 6.5620e-005 | 7.5044e-005 | 7.1224e-005 | 7.3518e-005 | 7.0841e-005 | 7.0469e-005 |
| 20. | 7.3691e-005 | 6.8486e-005 | 7.3838e-005 | 6.6253e-005 | 6.6767e-005 | 7.0925e-005 | 7.1028e-005 | 7.5410e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 21. | 7.5865e-005 | 7.2505e-005 | 7.6276e-005 | 7.0594e-005 | 6.8844e-005 | 6.7420e-005 | 6.8399e-005 | 6.8035e-005 |
| 22. | 7.3340e-005 | 7.3895e-005 | 7.2592e-005 | 6.5876e-005 | 7.0717e-005 | 6.5375e-005 | 6.5616e-005 | 6.5931e-005 |
| 23. | 6.3960e-005 | 6.2811e-005 | 6.2835e-005 | 6.3091e-005 | 6.5539e-005 | 6.4639e-005 | 6.3130e-005 | 6.2217e-005 |
| 24. | 6.1391e-005 | 6.5383e-005 | 6.9693e-005 | 6.6118e-005 | 6.1183e-005 | 6.0873e-005 | 6.6379e-005 | 6.4750e-005 |
| 25. | 6.9706e-005 | 6.5971e-005 | 6.8061e-005 | 6.6560e-005 | 6.8588e-005 | 6.6074e-005 | 6.6202e-005 | 6.6620e-005 |
| 26. | 6.3288e-005 | 5.9613e-005 | 6.1899e-005 | 5.8608e-005 | 6.0162e-005 | 6.2906e-005 | 6.3532e-005 | 6.0026e-005 |
| 27. | 5.9257e-005 | 5.8185e-005 | 5.8877e-005 | 6.0511e-005 | 5.7041e-005 | 5.6928e-005 | 5.7484e-005 | 5.9753e-005 |
| 28. | 6.2735e-005 | 5.8794e-005 | 6.1309e-005 | 5.8712e-005 | 5.8421e-005 | 5.8601e-005 | 5.7464e-005 | 5.8883e-005 |
| 29. | 6.7535e-005 | 6.4901e-005 | 6.2968e-005 | 6.4227e-005 | 6.3444e-005 | 6.2152e-005 | 6.3429e-005 | 6.2227e-005 |
| 30. | 7.1018e-005 | 7.0454e-005 | 6.9294e-005 | 6.8816e-005 | 6.8132e-005 | 6.6691e-005 | 7.0734e-005 | 6.5396e-005 |
| 31. | 6.4178e-005 | 6.5720e-005 | 5.9094e-005 | 6.3789e-005 | 6.3888e-005 | 6.3452e-005 | 6.4542e-005 | 6.4933e-005 |
| 32. | 6.3334e-005 | 6.1549e-005 | 6.2646e-005 | 6.4778e-005 | 6.2468e-005 | 6.4496e-005 | 6.2465e-005 | 6.1436e-005 |
| 33. | 6.4907e-005 | 6.9313e-005 | 6.3315e-005 | 6.2863e-005 | 6.5264e-005 | 6.4973e-005 | 6.2173e-005 | 6.1352e-005 |
| 34. | 6.5840e-005 | 6.4768e-005 | 6.3928e-005 | 6.3272e-005 | 6.0776e-005 | 6.0565e-005 | 5.9111e-005 | 6.0466e-005 |
| 35. | 6.3322e-005 | 6.2598e-005 | 6.3389e-005 | 6.3872e-005 | 6.2490e-005 | 6.1889e-005 | 6.1251e-005 | 5.8334e-005 |
| 36. | 6.7754e-005 | 6.0283e-005 | 5.9360e-005 | 7.1328e-005 | 6.8816e-005 | 6.4366e-005 | 6.0847e-005 | 6.0104e-005 |
| 37. | 6.1261e-005 | 6.1202e-005 | 6.0029e-005 | 6.0182e-005 | 6.0565e-005 | 6.3349e-005 | 6.1836e-005 | 6.3076e-005 |
| 38. | 6.3364e-005 | 6.1806e-005 | 6.4932e-005 | 6.8236e-005 | 6.7729e-005 | 6.3976e-005 | 6.2771e-005 | 6.3064e-005 |
| 39. | 6.2898e-005 | 6.3610e-005 | 6.6777e-005 | 6.1146e-005 | 6.5692e-005 | 6.4378e-005 | 6.5053e-005 | 6.1057e-005 |
| 40. | 7.0802e-005 | 6.9729e-005 | 7.0204e-005 | 6.3597e-005 | 6.2706e-005 | 7.5021e-005 | 6.8298e-005 | 6.6511e-005 |
| 41. | 6.6561e-005 | 6.4808e-005 | 6.3352e-005 | 6.2053e-005 | 6.3299e-005 | 6.6825e-005 | 6.9637e-005 | 6.6912e-005 |
| 42. | 7.0127e-005 | 6.5401e-005 | 6.7840e-005 | 6.2181e-005 | 6.6679e-005 | 6.6761e-005 | 6.6753e-005 | 7.1668e-005 |
| 43. | 7.2214e-005 | 6.5157e-005 | 6.3385e-005 | 7.0937e-005 | 6.3202e-005 | 7.1873e-005 | 6.6395e-005 | 6.9285e-005 |
| 44. | 6.2322e-005 | 6.1722e-005 | 6.5634e-005 | 6.4227e-005 | 6.0463e-005 | 5.8561e-005 | 5.8935e-005 | 6.1772e-005 |
| 45. | 6.1758e-005 | 6.4645e-005 | 6.1799e-005 | 6.1665e-005 | 6.0143e-005 | 6.0707e-005 | 6.1906e-005 | 6.1146e-005 |
| 46. | 6.8829e-005 | 6.6089e-005 | 6.5134e-005 | 6.5636e-005 | 6.7962e-005 | 6.4458e-005 | 6.4785e-005 | 6.8042e-005 |
| 47. | 6.4226e-005 | 6.2682e-005 | 6.6984e-005 | 6.3822e-005 | 6.6647e-005 | 5.9207e-005 | 5.9629e-005 | 5.9165e-005 |
| 48. | 6.5153e-005 | 6.0839e-005 | 6.3829e-005 | 6.5166e-005 | 6.3062e-005 | 6.3286e-005 | 6.2582e-005 | 6.3271e-005 |
| 49. | 6.6251e-005 | 6.5690e-005 | 5.9909e-005 | 6.3187e-005 | 6.5374e-005 | 6.0175e-005 | 6.4950e-005 | 6.7651e-005 |
| 50. | 6.5081e-005 | 6.1554e-005 | 5.9638e-005 | 5.9981e-005 | 5.8992e-005 | 6.1140e-005 | 5.8726e-005 | 6.3338e-005 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 51. | 6.0935e-005 | 5.9382e-005 | 6.0256e-005 | 6.1316e-005 | 6.0439e-005 | 6.1479e-005 | 5.8647e-005 | 5.9142e-005 |
| 52. | 6.0941e-005 | 6.0586e-005 | 6.0162e-005 | 6.5237e-005 | 6.3689e-005 | 6.3233e-005 | 6.1050e-005 | 5.9042e-005 |
| 53. | 6.3341e-005 | 6.4206e-005 | 6.6898e-005 | 6.0039e-005 | 6.3494e-005 | 6.4838e-005 | 6.2298e-005 | 5.9795e-005 |
| 54. | 6.6085e-005 | 6.8793e-005 | 6.4254e-005 | 6.4814e-005 | 6.5343e-005 | 6.8179e-005 | 6.9070e-005 | 6.9355e-005 |
| 55. | 6.5473e-005 | 6.4296e-005 | 6.6299e-005 | 6.3904e-005 | 6.9444e-005 | 6.6846e-005 | 6.2575e-005 | 6.2679e-005 |
| 56. | 6.9471e-005 | 6.6499e-005 | 6.7328e-005 | 6.4718e-005 | 6.5386e-005 | 6.3602e-005 | 6.4982e-005 | 6.7244e-005 |
| 57. | 6.7267e-005 | 5.8710e-005 | 5.9585e-005 | 6.0633e-005 | 5.9228e-005 | 6.8088e-005 | 6.7287e-005 | 5.9228e-005 |
| 58. | 6.6251e-005 | 6.5690e-005 | 5.9909e-005 | 6.3187e-005 | 6.5374e-005 | 6.0175e-005 | 6.4950e-005 | 6.7651e-005 |
| 59. | 6.5617e-005 | 5.9114e-005 | 5.9890e-005 | 5.9167e-005 | 5.8818e-005 | 6.0341e-005 | 6.0279e-005 | 5.9281e-005 |
| 60. | 6.0880e-005 | 6.2063e-005 | 6.2810e-005 | 5.7818e-005 | 5.8808e-005 | 6.3605e-005 | 5.8500e-005 | 5.7037e-005 |
| 61. | 6.2159e-005 | 6.4836e-005 | 6.4035e-005 | 5.8055e-005 | 5.7409e-005 | 5.8235e-005 | 5.7159e-005 | 5.9854e-005 |
| 62. | 6.1202e-005 | 6.3572e-005 | 5.9303e-005 | 6.0816e-005 | 6.1992e-005 | 5.8776e-005 | 5.9140e-005 | 6.1325e-005 |
| 63. | 6.4350e-005 | 5.8148e-005 | 5.7396e-005 | 6.2443e-005 | 5.9583e-005 | 6.3647e-005 | 6.4457e-005 | 6.4726e-005 |
| 64. | 6.4505e-005 | 6.2119e-005 | 6.2496e-005 | 6.3477e-005 | 6.8237e-005 | 5.9769e-005 | 6.6043e-005 | 6.1275e-005 |
| 65. | 6.3758e-005 | 6.3550e-005 | 6.6190e-005 | 6.0144e-005 | 6.3845e-005 | 6.3856e-005 | 6.1425e-005 | 6.3437e-005 |
| 66. | 6.2041e-005 | 6.1761e-005 | 5.9184e-005 | 6.2139e-005 | 5.9441e-005 | 5.9147e-005 | 5.8518e-005 | 6.0032e-005 |
| 67. | 6.5563e-005 | 6.0144e-005 | 6.0687e-005 | 6.0567e-005 | 6.3433e-005 | 7.0483e-005 | 6.4835e-005 | 6.0579e-005 |
| 68. | 6.4523e-005 | 6.6556e-005 | 6.8529e-005 | 5.9391e-005 | 6.1334e-005 | 5.9888e-005 | 6.0256e-005 | 6.0230e-005 |
| 69. | 6.2474e-005 | 6.2823e-005 | 6.5162e-005 | 6.2444e-005 | 6.5722e-005 | 6.1397e-005 | 5.9446e-005 | 5.9380e-005 |
| 70. | 6.3864e-005 | 6.7338e-005 | 6.7833e-005 | 6.6213e-005 | 6.5410e-005 | 6.1584e-005 | 6.3689e-005 | 6.7515e-005 |
| 71. | 5.9911e-005 | 5.8172e-005 | 5.7956e-005 | 5.7994e-005 | 6.1046e-005 | 6.0152e-005 | 5.7573e-005 | 6.2766e-005 |

# Appendix G

### The GCF for Dry , Oily, and Neutral Fingerprint Image

## For Neutral:

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 10.6390 | 10.6211 | 10.4639 | 10.0405 | 10.6695 | 10.7485 | 10.5718 | 10.4435 |
| 2. | 10.9799 | 11.6504 | 11.1250 | 10.6793 | 11.0022 | 11.0331 | 11.1509 | 10.8239 |
| 3. | 11.5778 | 11.0979 | 11.2354 | 10.7261 | 10.7776 | 10.1344 | 11.0445 | 11.2691 |
| 4. | 11.3768 | 10.9347 | 11.6558 | 11.8149 | 11.4606 | 11.7306 | 11.5973 | 11.6009 |
| 5. | 11.1903 | 11.4180 | 11.6687 | 10.3453 | 11.4020 | 11.7210 | 11.5738 | 10.2130 |
| 6. | 10.8715 | 12.1684 | 11.0328 | 12.2493 | 11.3218 | 10.8789 | 11.1841 | 11.8618 |
| 7. | 9.7700 | 10.4755 | 10.4021 | 10.8611 | 11.7306 | 10.9788 | 10.8981 | 11.9365 |
| 8. | 10.7929 | 11.4312 | 11.2163 | 11.3609 | 11.3206 | 11.2756 | 11.3760 | 11.4960 |
| 9. | 10.7651 | 10.8462 | 9.7592 | 10.5116 | 10.5449 | 9.8290 | 11.0740 | 11.1024 |
| 10. | 9.6204 | 10.0073 | 9.9176 | 9.5327 | 9.4757 | 9.6244 | 10.0335 | 9.5442 |
| 11. | 10.4987 | 11.5934 | 11.4924 | 11.1131 | 11.7088 | 11.7736 | 12.5077 | 11.6926 |
| 12. | 11.2721 | 11.6607 | 11.9491 | 11.6417 | 11.2899 | 11.5961 | 11.2608 | 11.2990 |
| 13. | 11.0803 | 11.6791 | 11.5030 | 11.4568 | 11.7148 | 11.5641 | 10.9571 | 11.4624 |
| 14. | 10.8777 | 11.2657 | 10.7127 | 11.2000 | 9.5449 | 9.4382 | 9.5271 | 9.4111 |
| 15. | 10.4669 | 11.1039 | 10.5497 | 10.5118 | 11.2517 | 11.1744 | 11.5591 | 10.9008 |
| 16. | 10.0110 | 11.3980 | 10.6226 | 11.3277 | 11.4834 | 11.3658 | 11.3273 | 11.5527 |
| 17. | 9.8789 | 9.2732 | 10.3908 | 10.8599 | 10.6735 | 11.1789 | 10.4625 | 10.8270 |
| 18. | 11.9326 | 12.1327 | 11.7650 | 12.0691 | 10.3191 | 12.1509 | 11.5347 | 11.3028 |
| 19. | 9.5684 | 10.5976 | 10.6809 | 10.5822 | 10.3863 | 10.0661 | 10.5195 | 9.4834 |
| 20. | 8.5192 | 9.7235 | 9.4246 | 9.4390 | 10.0198 | 9.6924 | 10.0374 | 9.2601 |
| 21. | 11.2409 | 11.3665 | 11.2416 | 10.3651 | 10.6312 | 10.6094 | 11.1231 | 11.3313 |
| 22. | 9.2876 | 10.1794 | 9.7093 | 10.1970 | 11.2119 | 10.5049 | 11.3189 | 10.7197 |
| 23. | 11.3041 | 11.2601 | 11.2346 | 11.0143 | 11.2599 | 11.2296 | 11.7772 | 11.0532 |
| 24. | 9.9861 | 10.9343 | 10.7499 | 10.7681 | 10.6652 | 10.0106 | 9.1223 | 10.7276 |
| 25. | 10.3370 | 10.6903 | 10.9059 | 10.7990 | 10.9064 | 10.1936 | 10.7795 | 10.5302 |
| 26. | 10.8122 | 11.5439 | 11.6064 | 11.1213 | 11.1230 | 10.5979 | 10.7279 | 10.9611 |
| 27. | 11.9139 | 11.6608 | 11.3607 | 11.2008 | 11.1753 | 11.4410 | 10.6403 | 10.3964 |
| 28. | 11.3551 | 10.5598 | 11.3847 | 10.3967 | 10.1108 | 10.1069 | 10.4963 | 10.2469 |
| 29. | 10.7019 | 11.6009 | 11.1607 | 11.7812 | 11.3977 | 11.4441 | 10.9373 | 10.8721 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 30. | 10.6456 | 11.3419 | 11.0037 | 11.8189 | 11.9925 | 11.8781 | 12.0641 | 10.5846 |
| 31. | 10.5577 | 10.4820 | 10.5087 | 10.1813 | 10.4407 | 10.4990 | 10.5577 | 10.8181 |
| 32. | 11.5065 | 11.9077 | 11.7577 | 11.5928 | 11.6872 | 11.6415 | 11.5224 | 10.7122 |
| 33. | 10.2426 | 11.0631 | 9.8698 | 11.1313 | 10.0691 | 10.5284 | 10.8421 | 11.3273 |
| 34. | 11.6158 | 11.2149 | 10.7779 | 11.1166 | 10.9550 | 10.9504 | 11.1607 | 10.9716 |
| 35. | 11.2856 | 10.6888 | 11.2911 | 10.9045 | 10.8524 | 11.6196 | 10.8723 | 10.7942 |
| 36. | 11.0236 | 10.3966 | 9.6220 | 9.9986 | 10.4211 | 10.9676 | 11.3669 | 10.8677 |
| 37. | 10.7675 | 11.2404 | 11.5816 | 11.8304 | 10.3811 | 11.1913 | 10.4849 | 10.5621 |
| 38. | 10.9640 | 11.2101 | 10.5592 | 11.0542 | 10.7702 | 11.1890 | 10.5010 | 10.5976 |
| 39. | 11.4766 | 11.7859 | 12.1069 | 11.9878 | 10.9265 | 12.1265 | 11.7670 | 11.1900 |
| 40. | 11.5073 | 10.4405 | 10.3182 | 10.7864 | 10.8393 | 10.6839 | 10.2941 | 10.4814 |
| 41. | 10.6074 | 11.3537 | 11.6377 | 11.2911 | 10.9666 | 11.5391 | 11.6935 | 11.8668 |
| 42. | 10.9403 | 11.4738 | 10.6418 | 10.2957 | 10.6061 | 10.5218 | 10.8911 | 10.9296 |
| 43. | 11.8915 | 11.8314 | 10.9328 | 10.8982 | 10.1371 | 10.5138 | 9.6155 | 10.2836 |
| 44. | 10.0088 | 10.4014 | 11.0673 | 10.3358 | 10.3775 | 11.2512 | 11.5509 | 11.0456 |
| 45. | 10.5244 | 11.1587 | 11.2993 | 11.4330 | 11.4424 | 11.5610 | 11.4787 | 11.7784 |
| 46. | 11.6178 | 11.7149 | 11.5769 | 11.8218 | 11.0960 | 11.2492 | 11.3054 | 11.6718 |
| 47. | 10.8835 | 11.1618 | 10.6930 | 10.6356 | 10.7133 | 10.7765 | 11.4380 | 11.2333 |
| 48. | 10.8580 | 10.4323 | 10.1884 | 10.6335 | 10.9611 | 10.8414 | 10.7625 | 11.2808 |
| 49. | 11.7234 | 11.5563 | 11.6673 | 11.4091 | 11.0139 | 11.6160 | 11.4302 | 11.8337 |
| 50. | 11.1366 | 11.0022 | 11.2787 | 11.4111 | 11.3545 | 11.2625 | 11.2716 | 11.4052 |
| 51. | 10.7931 | 11.9876 | 11.8385 | 11.7488 | 11.8522 | 11.5752 | 12.0988 | 11.7778 |
| 52. | 10.9409 | 9.6684 | 9.4562 | 9.2654 | 8.3234 | 9.4785 | 9.6046 | 9.2566 |
| 53. | 10.4667 | 10.0714 | 10.6377 | 11.0594 | 10.6159 | 11.3012 | 10.1038 | 10.3925 |
| 54. | 10.9351 | 10.8956 | 10.6627 | 11.0956 | 11.2669 | 10.4475 | 10.9608 | 10.9736 |
| 55. | 10.9963 | 11.1013 | 10.9904 | 10.6194 | 11.1877 | 11.1325 | 11.4788 | 11.2340 |
| 56. | 10.8886 | 10.8003 | 10.4210 | 10.4811 | 10.2349 | 11.2865 | 10.3589 | 10.5852 |
| 57. | 11.4510 | 11.2457 | 10.2421 | 10.8039 | 11.0932 | 11.3431 | 10.3199 | 11.1266 |
| 58. | 11.7234 | 11.5563 | 11.6673 | 11.4091 | 11.0139 | 11.6160 | 11.4302 | 11.8337 |
| 59. | 11.4355 | 10.7638 | 11.3313 | 11.1795 | 11.3449 | 11.1117 | 11.3020 | 11.1853 |
| 60. | 11.0943 | 11.0907 | 11.5271 | 11.1076 | 11.0453 | 11.1796 | 11.2849 | 11.2434 |
| 61. | 11.5517 | 11.5908 | 11.5087 | 11.0977 | 11.7016 | 10.7510 | 10.0463 | 10.0536 |
| 62. | 10.7899 | 9.9944 | 10.4685 | 9.5817 | 9.4376 | 9.5158 | 10.7428 | 10.7265 |
| 63. | 11.8180 | 11.5556 | 11.3714 | 11.6206 | 11.1937 | 11.3072 | 11.2843 | 11.3822 |
| 64. | 10.8000 | 11.6179 | 11.3497 | 11.0906 | 11.5448 | 11.4987 | 11.2092 | 11.1280 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 65. | 10.2179 | 9.8433 | 9.6333 | 10.4421 | 10.5646 | 10.4613 | 10.5440 | 10.3506 |
| 66. | 11.2464 | 11.3071 | 11.3718 | 9.6407 | 10.5476 | 9.7418 | 9.7458 | 9.9164 |
| 67. | 11.1377 | 11.3620 | 11.5990 | 11.6686 | 11.4330 | 11.4059 | 11.0622 | 11.2971 |
| 68. | 11.1509 | 11.6035 | 11.5541 | 11.6589 | 10.3695 | 11.7400 | 11.7543 | 12.1441 |
| 69. | 11.0372 | 10.9782 | 11.6948 | 11.6234 | 11.3946 | 11.7143 | 11.3334 | 11.5582 |
| 70. | 10.5807 | 11.3878 | 10.9362 | 11.3775 | 10.7034 | 10.9980 | 11.1547 | 11.1587 |
| 71. | 11.0582 | 11.3441 | 11.8217 | 11.6989 | 11.1354 | 11.3117 | 11.5960 | 11.2843 |

## For Dry :

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 10.8909 | 10.6000 | 9.9507 | 8.6157 | 10.0419 | 10.2489 | 10.0785 | 10.2556 |
| 2. | 10.3194 | 10.4392 | 10.3576 | 10.4759 | 10.3921 | 10.6078 | 10.4114 | 10.7086 |
| 3. | 9.8998 | 10.0684 | 9.9692 | 10.3998 | 10.1656 | 10.9932 | 9.5940 | 10.1152 |
| 4. | 10.8721 | 10.0941 | 10.8423 | 10.7646 | 10.3747 | 10.4062 | 10.0120 | 9.7761 |
| 5. | 10.4976 | 10.8023 | 10.2852 | 10.5508 | 10.0732 | 10.9514 | 9.8990 | 10.1501 |
| 6. | 10.0977 | 10.8719 | 10.7689 | 10.4830 | 10.8141 | 10.4288 | 10.6711 | 10.8141 |
| 7. | 10.4341 | 10.7799 | 10.4597 | 10.4300 | 10.1264 | 10.5147 | 10.8478 | 10.3680 |
| 8. | 10.0303 | 10.4145 | 10.8862 | 10.3154 | 10.8768 | 10.1876 | 11.0537 | 10.6165 |
| 9. | 10.9728 | 10.5316 | 10.3845 | 10.4640 | 10.8859 | 10.1927 | 10.8151 | 10.4581 |
| 10. | 10.1151 | 10.0447 | 9.3459 | 9.4211 | 10.2356 | 9.0985 | 9.4752 | 10.1250 |
| 11. | 10.5539 | 10.5851 | 10.0318 | 10.8335 | 10.1276 | 10.8740 | 11.9084 | 10.5125 |
| 12. | 10.3385 | 10.4794 | 10.7087 | 10.6003 | 10.6517 | 10.9471 | 11.4820 | 10.7668 |
| 13. | 10.3381 | 10.3169 | 10.2262 | 10.3088 | 10.3544 | 10.4580 | 10.2960 | 10.0420 |
| 14. | 10.9901 | 10.6855 | 10.4621 | 10.5050 | 10.5447 | 10.5099 | 10.3944 | 10.7012 |
| 15. | 10.5755 | 10.3139 | 10.1326 | 10.0312 | 10.3217 | 10.4291 | 10.2022 | 10.7111 |
| 16. | 11.4838 | 11.0184 | 10.9763 | 9.1233 | 10.3267 | 10.9059 | 9.3310 | 10.1765 |
| 17. | 10.1833 | 9.2845 | 9.5333 | 10.1275 | 10.5987 | 9.6037 | 9.7958 | 10.5999 |
| 18. | 10.0510 | 10.7700 | 10.7234 | 10.5714 | 10.8579 | 10.8936 | 10.1094 | 10.7956 |
| 19. | 10.3295 | 10.3131 | 10.0768 | 9.7476 | 9.6281 | 9.3931 | 9.6842 | 10.4817 |
| 20. | 9.4012 | 10.0770 | 10.4968 | 9.8341 | 9.8426 | 9.8087 | 9.3791 | 9.0831 |
| 21. | 10.4346 | 10.8673 | 10.9819 | 10.0400 | 10.4965 | 10.3449 | 11.4018 | 10.1723 |
| 22. | 9.4069 | 9.6394 | 9.3554 | 9.4708 | 10.4410 | 10.2846 | 9.5871 | 10.2432 |
| 23. | 10.1078 | 10.1281 | 10.4516 | 10.4765 | 10.9693 | 10.1924 | 10.4488 | 10.3672 |
| 24. | 10.0539 | 10.0799 | 9.6989 | 10.1279 | 10.6708 | 9.5911 | 9.7511 | 10.3560 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 25. | 10.9007 | 10.6766 | 10.7413 | 9.2283 | 10.7915 | 10.7182 | 9.9875 | 10.6636 |
| 26. | 11.1313 | 10.1922 | 9.9946 | 9.6228 | 10.1718 | 10.4422 | 10.2253 | 10.5680 |
| 27. | 10.3328 | 10.5630 | 10.1654 | 10.0252 | 10.4972 | 9.6099 | 10.4326 | 10.4432 |
| 28. | 10.3433 | 10.6387 | 10.4569 | 10.0878 | 10.5220 | 10.0257 | 10.1928 | 10.9424 |
| 29. | 10.0277 | 9.3581 | 9.4725 | 10.1184 | 10.2990 | 10.5680 | 10.9628 | 9.9936 |
| 30. | 9.6233 | 9.8164 | 10.0180 | 10.3846 | 10.1965 | 9.8742 | 10.2701 | 10.8806 |
| 31. | 10.1448 | 10.0747 | 10.9238 | 9.7299 | 10.2555 | 10.7306 | 10.9872 | 10.5500 |
| 32. | 10.7179 | 9.3921 | 10.0465 | 9.3599 | 9.5552 | 10.5427 | 10.0830 | 10.2142 |
| 33. | 10.3452 | 10.2751 | 9.8832 | 10.8781 | 10.2391 | 10.0940 | 10.8482 | 10.1642 |
| 34. | 10.7514 | 11.0762 | 11.1554 | 11.1931 | 11.2712 | 10.4910 | 10.8074 | 11.1375 |
| 35. | 10.7849 | 10.5897 | 10.3910 | 10.0365 | 10.7627 | 10.7227 | 10.3875 | 10.7961 |
| 36. | 10.6258 | 10.0953 | 9.5703 | 10.4843 | 9.6273 | 10.2984 | 10.3296 | 9.6941 |
| 37. | 10.0211 | 10.4477 | 10.4446 | 10.2412 | 10.2509 | 10.0487 | 10.1552 | 10.0765 |
| 38. | 9.6903 | 10.0534 | 9.4619 | 10.2078 | 9.2506 | 10.5904 | 10.6938 | 9.9487 |
| 39. | 10.5317 | 10.6609 | 10.3990 | 10.2918 | 10.0924 | 9.8486 | 10.1770 | 10.0510 |
| 40. | 10.7063 | 10.9206 | 10.4957 | 10.9806 | 10.9563 | 10.5990 | 10.2698 | 10.1147 |
| 41. | 10.5484 | 10.3866 | 10.5667 | 10.5165 | 10.1051 | 10.8928 | 10.8768 | 10.4860 |
| 42. | 9.3368 | 10.6266 | 10.1614 | 10.1795 | 9.9493 | 10.2718 | 10.3491 | 10.5397 |
| 43. | 9.6838 | 10.1524 | 10.1938 | 11.0642 | 9.9400 | 10.0437 | 9.7553 | 10.9827 |
| 44. | 10.0729 | 10.0009 | 10.0951 | 10.7562 | 10.5153 | 10.8905 | 11.4342 | 11.2093 |
| 45. | 9.9835 | 11.0439 | 10.9099 | 11.2192 | 10.8536 | 10.0290 | 10.2962 | 10.8500 |
| 46. | 10.5835 | 10.1765 | 10.4844 | 10.6320 | 10.5959 | 10.6846 | 10.3729 | 10.8180 |
| 47. | 10.0504 | 10.1820 | 10.2355 | 10.4254 | 10.4957 | 10.3581 | 10.0616 | 10.2621 |
| 48. | 10.1508 | 10.0902 | 10.3058 | 10.6985 | 10.2580 | 10.0948 | 10.7520 | 10.8441 |
| 49. | 10.3833 | 10.5921 | 10.6282 | 10.6318 | 10.6291 | 10.5260 | 10.2370 | 10.5009 |
| 50. | 10.1456 | 10.8510 | 10.7709 | 10.5735 | 10.4807 | 10.2331 | 10.2011 | 10.3508 |
| 51. | 10.2952 | 10.0070 | 10.0408 | 10.6783 | 10.3660 | 10.9266 | 10.9586 | 10.7826 |
| 52. | 9.7914 | 10.1361 | 9.9725 | 9.7577 | 9.9029 | 10.0608 | 10.0342 | 10.1097 |
| 53. | 10.0680 | 10.7990 | 9.8243 | 9.6422 | 10.1795 | 10.7702 | 10.3268 | 10.2728 |
| 54. | 10.4771 | 10.1910 | 10.3234 | 10.2479 | 10.3365 | 10.2494 | 10.3966 | 10.5077 |
| 55. | 10.9224 | 10.2677 | 10.2677 | 10.4987 | 10.3169 | 10.0742 | 10.2115 | 10.9993 |
| 56. | 10.2652 | 10.3457 | 10.3129 | 10.1121 | 10.7612 | 10.1326 | 10.6604 | 10.9274 |
| 57. | 10.6808 | 10.5449 | 10.5008 | 9.8511 | 10.7835 | 10.4468 | 10.5759 | 10.7026 |
| 58. | 10.3833 | 10.5921 | 10.6282 | 10.6318 | 10.6291 | 10.5260 | 10.2370 | 10.5009 |
| 59. | 10.0708 | 10.0299 | 10.3719 | 10.2216 | 10.9746 | 10.2579 | 10.9296 | 10.2476 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 60. | 10.7011 | 10.3953 | 10.2211 | 10.4100 | 10.7064 | 10.9088 | 10.1002 | 10.8322 |
| 61. | 10.8779 | 10.2730 | 10.4711 | 10.3645 | 10.8679 | 10.1070 | 10.2019 | 10.0351 |
| 62. | 9.9197 | 10.1146 | 10.9086 | 10.5542 | 10.1774 | 10.7082 | 10.6236 | 10.6047 |
| 63. | 10.4108 | 10.0721 | 10.0969 | 10.0420 | 10.1004 | 10.9969 | 10.0203 | 10.1781 |
| 64. | 10.0302 | 10.6337 | 10.6063 | 10.5541 | 10.7037 | 10.4258 | 10.7062 | 10.2009 |
| 65. | 10.3221 | 10.6295 | 10.5965 | 10.6426 | 10.4522 | 10.4579 | 10.5387 | 9.9326 |
| 66. | 10.4147 | 10.1709 | 10.7493 | 10.0657 | 10.8296 | 10.0812 | 10.8265 | 10.9254 |
| 67. | 10.4271 | 10.4019 | 10.4646 | 10.0875 | 10.3993 | 10.4550 | 10.3095 | 10.2074 |
| 68. | 10.5056 | 10.3500 | 10.9298 | 10.8010 | 10.1447 | 10.8462 | 10.3694 | 10.6334 |
| 69. | 10.8906 | 10.0543 | 10.4327 | 10.9568 | 10.1721 | 10.3313 | 10.4082 | 10.4361 |
| 70. | 11.2044 | 10.9619 | 10.1789 | 11.3963 | 11.7074 | 10.6869 | 10.3548 | 11.2602 |
| 71. | 10.7952 | 10.6714 | 10.2720 | 10.6165 | 10.5764 | 10.2940 | 10.5177 | 10.4205 |

## For Oily:

| Finger# / Person# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1. | 8.8731 | 8.0432 | 8.8731 | 9.1062 | 9.6690 | 9.4693 | 9.1112 | 9.4580 |
| 2. | 9.7102 | 9.7663 | 9.2221 | 9.3023 | 9.4871 | 9.3344 | 9.5642 | 9.3806 |
| 3. | 9.1863 | 9.1500 | 9.5007 | 9.3800 | 9.5095 | 9.3179 | 9.0991 | 9.3941 |
| 4. | 9.0492 | 9.4927 | 9.3038 | 9.7260 | 9.3645 | 9.5527 | 9.3402 | 9.7522 |
| 5. | 8.9457 | 8.7044 | 9.5859 | 9.4887 | 8.4590 | 10.0956 | 8.9676 | 9.5749 |
| 6. | 9.3100 | 8.9702 | 9.3109 | 9.7554 | 9.7326 | 8.7070 | 8.8463 | 8.1524 |
| 7. | 9.4839 | 9.5207 | 9.5214 | 9.3407 | 9.7917 | 9.7056 | 9.4569 | 9.3121 |
| 8. | 9.4562 | 9.8937 | 9.3447 | 9.1909 | 9.6389 | 9.6456 | 9.6587 | 9.3711 |
| 9. | 9.7015 | 9.2267 | 9.4045 | 9.0213 | 9.7878 | 9.4134 | 9.9440 | 9.0821 |
| 10. | 9.6244 | 9.9517 | 9.4136 | 9.3208 | 9.7855 | 9.3969 | 9.6940 | 9.4204 |
| 11. | 9.2379 | 9.3942 | 9.2297 | 9.5418 | 9.9987 | 9.2804 | 9.1404 | 9.0172 |
| 12. | 9.0446 | 9.7311 | 9.8107 | 9.1959 | 9.4493 | 9.0281 | 9.7015 | 9.6101 |
| 13. | 9.3290 | 9.1707 | 9.6259 | 9.7526 | 9.9486 | 9.7145 | 9.2677 | 9.3992 |
| 14. | 8.4995 | 8.6661 | 8.6752 | 8.4235 | 9.0314 | 8.9615 | 9.0039 | 8.8471 |
| 15. | 9.1373 | 9.8026 | 9.1764 | 9.1154 | 9.1224 | 9.5478 | 9.5448 | 9.6958 |
| 16. | 9.8896 | 9.0638 | 9.1756 | 9.8762 | 9.4890 | 9.7889 | 9.5589 | 9.7258 |
| 17. | 11.6605 | 9.3357 | 9.3229 | 9.7756 | 10.9932 | 11.5310 | 11.1101 | 11.6612 |
| 18. | 9.5179 | 10.1771 | 11.1575 | 11.9875 | 11.4535 | 10.5284 | 11.0697 | 9.8869 |
| 19. | 8.3098 | 8.3851 | 9.4622 | 8.1487 | 9.0571 | 8.8797 | 8.5845 | 9.4298 |
| 20. | 8.9450 | 9.9200 | 9.2126 | 9.7131 | 9.8773 | 9.2503 | 9.3094 | 9.2930 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 21. | 9.1522 | 9.1382 | 9.0139 | 9.7203 | 9.1028 | 9.0912 | 9.7979 | 9.2406 |
| 22. | 8.9008 | 8.5759 | 8.6442 | 9.4546 | 8.9795 | 9.2894 | 8.8327 | 9.5754 |
| 23. | 8.4393 | 8.7705 | 8.4477 | 8.3697 | 8.1782 | 8.1210 | 8.5770 | 8.6209 |
| 24. | 9.4162 | 9.9254 | 9.4092 | 9.0391 | 9.1887 | 9.6193 | 9.1386 | 9.7920 |
| 25. | 8.7899 | 9.1122 | 9.6466 | 9.8592 | 9.7010 | 8.9831 | 9.4058 | 9.0562 |
| 26. | 9.7498 | 9.4617 | 9.9629 | 9.6772 | 9.9866 | 9.1754 | 9.1204 | 9.8120 |
| 27. | 10.3639 | 10.0907 | 10.1090 | 9.7729 | 9.9192 | 10.6714 | 10.9297 | 10.5176 |
| 28. | 9.1454 | 9.3869 | 9.8795 | 9.0676 | 9.3035 | 8.8866 | 9.5176 | 8.7703 |
| 29. | 8.0735 | 9.5713 | 9.5299 | 9.9542 | 9.4582 | 9.6676 | 9.9698 | 9.6246 |
| 30. | 9.4650 | 9.5889 | 9.5950 | 9.4454 | 9.5111 | 9.0151 | 9.0775 | 9.1013 |
| 31. | 9.1258 | 9.3126 | 9.4475 | 9.6306 | 9.7424 | 9.7861 | 9.4552 | 9.3301 |
| 32. | 9.1572 | 8.7450 | 9.2560 | 8.8619 | 9.4715 | 8.7991 | 9.6340 | 9.3414 |
| 33. | 9.1460 | 8.9548 | 8.6838 | 8.6939 | 9.8484 | 9.3433 | 9.1969 | 10.9183 |
| 34. | 9.7336 | 9.4494 | 9.9968 | 9.1550 | 9.1533 | 9.4583 | 9.3851 | 11.6447 |
| 35. | 9.2432 | 9.2098 | 9.7664 | 10.6352 | 9.3294 | 9.4062 | 9.2521 | 9.9273 |
| 36. | 9.1615 | 9.5084 | 9.2460 | 9.2958 | 9.8050 | 9.1561 | 9.6768 | 9.2045 |
| 37. | 8.8330 | 9.2603 | 8.6501 | 8.8927 | 8.7611 | 9.7790 | 9.2333 | 9.6497 |
| 38. | 9.1405 | 9.7923 | 9.4171 | 10.8762 | 9.5552 | 9.3442 | 9.8132 | 9.8347 |
| 39. | 9.3383 | 9.1746 | 9.9659 | 9.7543 | 9.0242 | 9.5798 | 9.4927 | 9.3707 |
| 40. | 9.6220 | 8.5578 | 8.3892 | 8.2554 | 8.6089 | 9.5762 | 8.8911 | 8.7535 |
| 41. | 9.1798 | 9.0788 | 9.7600 | 9.7999 | 9.0469 | 9.4159 | 9.7065 | 9.0088 |
| 42. | 9.1622 | 9.5860 | 9.1741 | 8.9565 | 9.8003 | 9.7351 | 9.8855 | 9.1989 |
| 43. | 9.4980 | 9.3635 | 9.0114 | 9.1238 | 9.5053 | 9.0435 | 9.6933 | 9.2831 |
| 44. | 9.8033 | 9.7400 | 9.6908 | 9.5158 | 9.6830 | 9.7946 | 9.6397 | 9.0876 |
| 45. | 9.7149 | 9.2428 | 9.3877 | 9.5541 | 9.3100 | 9.5069 | 9.2711 | 8.9906 |
| 46. | 8.3247 | 8.4384 | 8.3613 | 8.5317 | 8.8534 | 9.3788 | 9.0555 | 9.0379 |
| 47. | 8.3882 | 10.4171 | 8.5410 | 9.3011 | 9.9368 | 8.9739 | 10.7028 | 9.5669 |
| 48. | 9.9968 | 9.3795 | 9.0314 | 9.0352 | 9.8926 | 9.3052 | 9.3210 | 9.6837 |
| 49. | 10.3034 | 10.8470 | 11.4184 | 10.4999 | 10.6979 | 10.8613 | 10.5744 | 9.0086 |
| 50. | 9.3351 | 9.4793 | 9.6247 | 9.2587 | 9.3537 | 9.4767 | 9.2337 | 9.2759 |
| 51. | 9.7315 | 9.4136 | 9.1285 | 9.0803 | 9.1437 | 9.1456 | 9.6709 | 9.8780 |
| 52. | 9.3785 | 9.6007 | 9.7555 | 8.3820 | 9.5882 | 9.8967 | 9.2796 | 9.5913 |
| 53. | 9.5960 | 9.2058 | 9.9530 | 10.0587 | 9.2674 | 9.1159 | 9.5672 | 9.2682 |
| 54. | 9.9116 | 9.5234 | 9.2550 | 9.9168 | 9.9860 | 9.1020 | 9.2268 | 9.5964 |
| 55. | 9.2173 | 9.8772 | 9.6634 | 9.0624 | 9.0481 | 9.6052 | 9.7572 | 9.8794 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 56. | 9.2935 | 9.3960 | 9.6666 | 9.5684 | 8.9737 | 9.9675 | 9.6596 | 8.6583 |
| 57. | 9.6140 | 9.2771 | 9.0350 | 9.6672 | 9.2678 | 9.6450 | 9.7779 | 9.2678 |
| 58. | 9.3034 | 9.8470 | 9.4184 | 9.4999 | 9.6979 | 9.8613 | 9.5744 | 9.0086 |
| 59. | 9.5057 | 9.4674 | 9.6785 | 9.5344 | 9.9658 | 9.0066 | 9.0989 | 9.8421 |
| 60. | 9.8333 | 9.9962 | 9.9146 | 9.0489 | 9.0620 | 9.0837 | 10.6806 | 10.2772 |
| 61. | 9.4780 | 9.5483 | 9.6244 | 9.4184 | 9.4460 | 9.9293 | 9.7363 | 11.6942 |
| 62. | 9.3252 | 9.2540 | 9.9667 | 9.2386 | 9.1047 | 9.4884 | 9.0349 | 10.8701 |
| 63. | 8.4570 | 9.9554 | 9.2204 | 9.6258 | 9.3726 | 9.9288 | 9.1634 | 9.4904 |
| 64. | 9.3692 | 9.6890 | 9.7206 | 9.7652 | 9.3338 | 9.3796 | 9.1834 | 9.0442 |
| 65. | 9.4362 | 9.3298 | 9.2502 | 9.5443 | 9.3693 | 9.3727 | 9.5482 | 9.6491 |
| 66. | 9.1802 | 9.7091 | 9.4486 | 9.8222 | 9.6251 | 9.1830 | 9.8974 | 9.1789 |
| 67. | 9.2350 | 9.6354 | 9.3666 | 9.1769 | 9.5598 | 9.5567 | 9.0890 | 9.4067 |
| 68. | 8.2437 | 9.6485 | 9.5185 | 9.9118 | 9.9522 | 9.6986 | 9.5421 | 9.0128 |
| 69. | 9.8737 | 9.8171 | 9.5528 | 9.0698 | 9.3712 | 9.6999 | 9.6559 | 9.6042 |
| 70. | 9.3620 | 9.0114 | 9.6350 | 9.7252 | 9.6424 | 9.1698 | 9.2514 | 9.7787 |
| 71. | 9.7197 | 9.1626 | 9.1734 | 9.5404 | 9.0611 | 9.9417 | 9.8403 | 9.6974 |

# List of Publications

1. Fingerprint Image Enhancement and Quality Analysis – A Survey- in International Journal of Computer Networks and Communications Security· September 2014.
2. Fingerprint Image Quality Analysis Using Fuzzy Logic Technique- in International Arab Conference on Information Technology (ACIT'2015) December 2015.
3. Fingerprint Image Quality Fuzzy System in International Arab Journal of Information Technology, February 2016.
4. Dry Fingerprint Image Enhancement using Fuzzy Morphology - in International Journal of Knowledge Engineering and Data Mining (IJKEDM). (2016).
5. Oily Fingerprint Image Enhancement using Fuzzy Morphology in the proceedings of the International Arab Conference on Information Technology (ACIT'2016).
6. Fingerprint Intelligent System for Servicing the Holy Sites in Saudi Arabia in the International Conference on Islamic Applications in Computer Science and Technologies (IMAN 2016).
7. Localization Of Pilgrims In El Hajj Using Fuzzy Logic In Mobile Sensor Networks in the International Conference on Islamic Applications in Computer Science and Technologies (IMAN 2016).
8. Android App for Muslim Daily Activities in the International Conference on Islamic Applications in Computer Science and Technologies (IMAN 2016).