FINGERPRINT QUALITY ESTIMATION AND MATCHING ALGORITHMS

By

Muhammad Umar Munir
(2005-NUST-PhD-CSE-03)

Advisor

Professor Dr Muhammad Younus Javed

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

2012

FINGERPRINT QUALITY ESTIMATION AND MATCHING ALGORITHMS

by

MUHAMMAD UMAR MUNIR

A dissertation submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSPHY

in

COMPUTER SOFTWARE ENGINEEING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

2012

Abstract

Fingerprint Quality Estimation and Matching Algorithms

Accurate and reliable automatic personal identification is critical in wide range of application domains such as National ID card, Electronic Commerce, ATMs etc. Biometrics which refers to automatic identification of a person based on his physiological or behavioral characteristics is inherently more reliable, in differentiating an authorized person from an imposter, than traditional password and PIN number based methods. Among all the biometric techniques, fingerprint based authentication is mostly used because of its reliability, low cost and ease of integration.

The performance of automatic fingerprint identification systems relies heavily on the quality of fingerprint images. The quality of fingerprint image effects the accurate extraction of minutiae. The low quality image contains large number of false minutiae as compared to good quality image. In this dissertation, a novel technique has been proposed for quality estimation of fingerprint images. A set of statistical and frequency features has been calculated from the fingerprint image. K-means clustering algorithm has been utilized to classify the fingerprint image into four classes i.e. good, dry, normal and wet. It has been shown through experimental results that the performance of minutiae based matcher is improved when the quality of fingerprint image is incorporated in the matching stage.

A novel fingerprint matching algorithm based on Zernike moments is also proposed in this dissertation. For fingerprint matching, it is desirable to obtain a fingerprint representation invariant to translation and rotation. Translation invariance is achieved by transforming the fingerprint image into frequency domain and taking the absolute yielding the spectrum of an image invariant to translation. For rotation invariance, Zernike moments are calculated which are invariant to rotation. The fingerprint image is first enhanced and then converted into frequency domain by taking its Discrete Fourier Transform. Then the magnitude of the Zernike moments is calculated. The fingerprint matching is based on the normalized Euclidean distance between the two corresponding Zernike moments of stored template and query fingerprint image. Experimental results show that the proposed method has better performance in terms of matching accuracy as compared to the traditional Gabor filter based fingerprint matching methods.

Acknowledgment

Praise to Almighty Allah for bestowing upon me strength and knowledge, to conclude this aspiration in time and craft a substantial contribution.

I would like to acknowledge all the people who have assisted me during my PhD study in Computer Software Engineering at College of Electrical and Mechanical Engineering, National University of Sciences and Technology, Rawalpindi. I am most grateful to my adviser, Brig. Dr. Muhammad Younus Javed, for his professional and personal advice, help and guidance. He has been very understanding and supportive. I am very fortunate to have him as an adviser. I would like to especially thank Dr. Shoab Ahmad Khan for his valuable advice, help and suggestions. Many thanks to Dr. Ejaz Muhammad, Dr. Shoab Ahmad Khan and Dr Javaid Iqbal for their willingness to serve on my GEC committee. I am grateful to HQ NUST, National ICT R&D Fund and HEC for providing research funding through various projects which benefitted me in my doctoral research work.

My fellow students and colleagues have provided help and moral support throughout my stay in E&ME College. I would like to thank them for their interest and concern, especially Fayyaz Rafiq, Shariyar, and Dr. Saad Rehman.

My sincere thanks go to my mother for her never-fading love, care, understanding and encouragement. I could not have accomplished anything without her love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Biometrics

Biometrics is the method of identifying a person by his physiological or behavioral characteristics. It has the ability to differentiate between the genuine person and an imposter. Physiological characteristics include fingerprints, face, iris, retina etc. The behavioral characteristics include signature, gait, key stroke dynamics etc. Fingerprint and face recognition are the widely used biometric technologies in today's world [1][2][3]. These biometrics technologies are currently installed in ATMs, Airports, Border Access Points and Military Installations etc. These technologies are more reliable and secure than password based systems as passwords can be cracked by an imposter. The physiological and behavioral characteristics are unique to every individual and cannot be replicated easily [4][5][6]. Biometrics has a disadvantage also. One main disadvantage is that once the biometric characteristics are compromised than it cannot be easily replaced whereas if the password is cracked by an imposter, it can be changed easily. Among all the biometrics (e.g., face, fingerprint, hand geometry, iris, retina, signature, voice print, hand vein, gait, ear, odor, keystroke dynamics, etc.), fingerprint based biometrics is one of the most reliable, mature and established technique.

Figure 1.1 shows a generic biometric system. The sensor which is a face or fingerprint scanner acquires an image of face or fingerprint. Then the biometric system preprocesses the acquired data, i.e. a fingerprint image. In preprocessing, a fingerprint image is segmented and enhanced. Then the features are extracted from a fingerprint image and stored in a template. The fingerprint matcher compares the features of query fingerprint image with the stored templates and based on some threshold value, it is decided that query fingerprint image is matched or not.

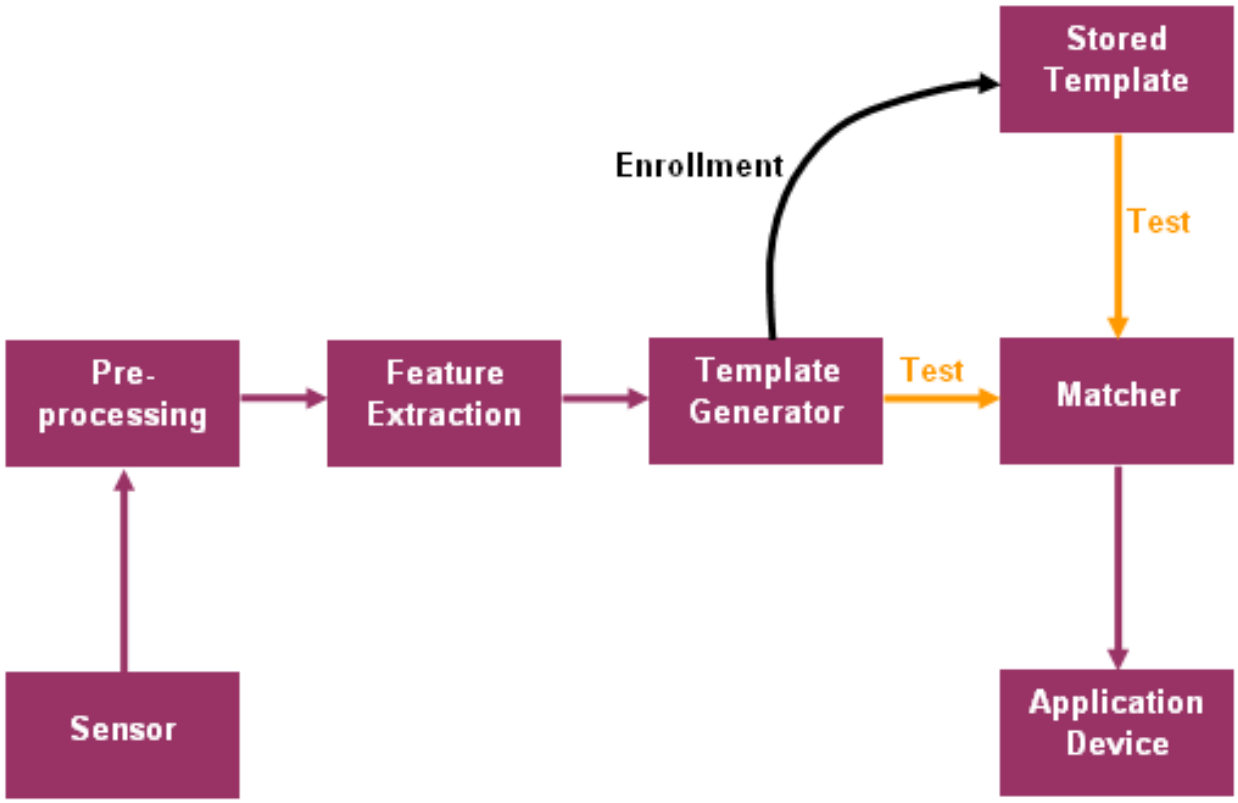

Figure 1.1: A Generic Biometric System

1.2 Biometrics Modes

Any biometrics system operates in one of two modes [7][8].

a) Verification mode

b) Identification mode

The mode in which biometric system is operating depends upon the requirement of the application.

1.2.1 Verification Mode

In the verification mode, the user enters his user name and presents his biometrics e.g. face or fingerprint as password. The biometrics system searches for the user name and if the user name is found in the database then the biometrics data (face or fingerprint) presented by the user is compared with the stored biometrics data. If the user is genuine person, then the biometrics system successfully matches the biometrics data. If the user is an imposter, then the biometrics data is not matched. In this mode, only *one to one* matching is performed. This requires very less amount of time for matching as matcher is run only once.

1.2.2 Identification Mode

In identification mode, user only presents his biometrics e.g. face, fingerprint, iris etc. The biometrics system compares the biometric data of the user with all the biometrics data stored in the database. In this case, if the biometrics data of the user matches with the stored data, then the user is successfully identified. In this mode, *one to many* matching is performed. This requires a

large amount of time for matching as the matcher compares the presented biometrics data of user with the biometrics data of all the users stored in the database until the match is found. . The matching algorithms used in identification mode should be very efficient as thousands of matches have to be performed to identify a person.

1.3 Biometric Applications

Biometrics has been evolving very rapidly and widely used in today's era. There are forensics applications such as criminal identification and verification. It has utilization in banking industry, e-commerce applications, secure military installations, computerized identification cards, passports and border access control systems etc. There are other biometrics applications which include timing in/out and attendance of employees in an organization. Electronic banking is one of the most important and emerging biometrics application area as there is a rapid growth in electronic transactions. Now a days, ATM cards have a built-in biometrics features. The biometrics data of a user has been stored in the memory of ATM card. When the user wants any electronics transaction through his ATM card, the ATM system asks the user to verify through his face or fingerprint. The user presents his face or fingerprint in front of scanner and then the system verifies the user by comparing his biometrics data with the data stored on card. If the user is genuine, then he is allowed to do transactions.

1.4 Biometric Technologies

There are various types of biometrics technologies and their use is dependent upon the requirement of the specific application. For example, in secure military installations, a combination of biometrics technologies which include face recognition, iris recognition and fingerprint identification can be utilized for identity verification of a person. By combining these technologies, the error rate can be reduced substantially and reliability and performance can be increased [9][10]. For optimal performance, multi-biometrics systems can be utilized whereas a single biometrics system cannot guarantee a minimum error rate and maximum reliability. The most commonly used biometrics technologies are briefly explained below:

1.4.1 Face biometrics

Face biometrics and their use can be greatly influenced by the conditions in which they are used [3][8]. As such, this biometric would be used mainly in offices with generally acceptable lighting conditions. The user would normally be seated at his/her desk for verification and would generally be authenticating based on a claim of identity. This claim of identity could take the form of inserting a smart card or providing a user ID. In addition, the user would authenticate at least three to four times a day. The speed of the authentication would need to be sufficient to make it usable, and it is quite possible that the facial expression would not always be the same. With these as parameters, each algorithm is evaluated as to its suitability for this environment.

1.4.1.1 Eigenface

The Eigenface algorithm is relatively quick with its searches. It generally requires good lighting and the face to be presented in a full frontal orientation. It does not deal well with variations in facial expression. As such, the user would need to present a face that is always non-expressive. This may not always be possible. Additionally, it is not consistent with people who sometimes wear glasses or grow beards. This may require the user to be re-enrolled if the user wears glasses all the time have a beard. Eigenface is a good general-purpose algorithm to use when the user community is relatively controlled and conditioned. Eignfaces are shown in Figure 1.2. It may not be appropriate for less structured user communities. The fact that wearing glasses and beard growth affects the algorithm's ability to authenticate could lead to a large false reject rate (FRR). This in turn could lead to more calls to the help desk and reduced user satisfaction.

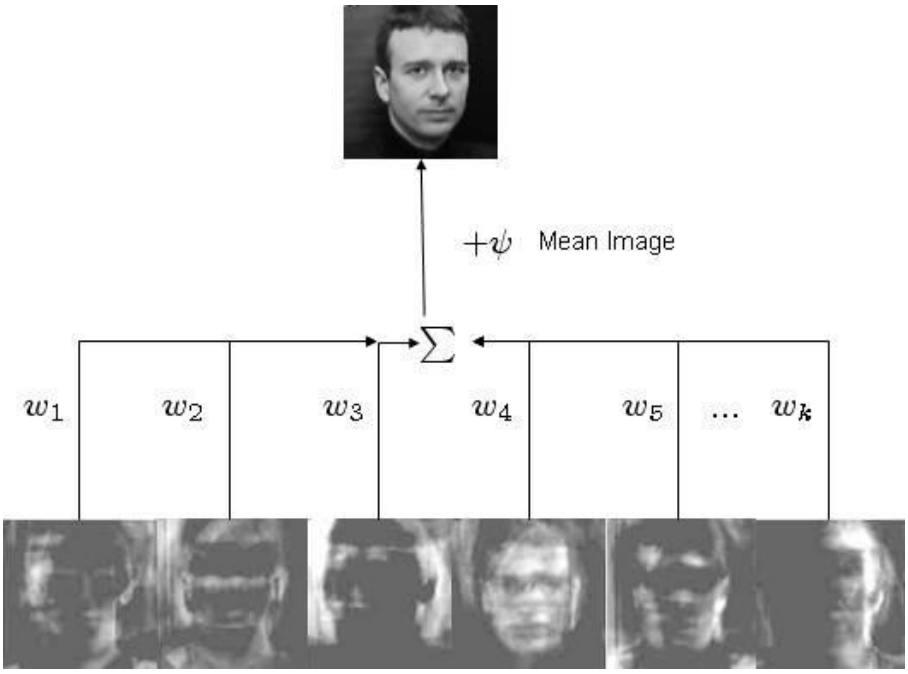

Figure 1.2: Face recognition using Eignfaces

1.4.1.2 Local feature analysis

Local feature analysis uses macro facial features along with bone structure and changes in shading to define anchor points. As such, it is much more forgiving for less than ideal lighting conditions and users who sometimes wear glasses and grow beards. It can also tolerate the face not being presented in a full frontal view. Since the macro features and bone structure are used as the anchor points, the head does not have to be held still for imaging to occur. In general, local feature analysis is very suitable to use as a network security biometric.

1.4.1.3 Neural network

Neural network uses a learning method to teach the network how to recognize and differentiate the face. As such, it is very good at isolating the face from a complex background. While some office areas may seem more "jungle-like" than others, in general, office environments are relatively uncluttered. The algorithm also requires a large database of images to get bootstrapped, and therefore, can be slower in processing the requested face for authentication. Additionally, the neural network requires a full frontal view of the face with good lighting. This requirement is within our parameters of the office environment. Generic neural network based face recognition is shown in Figure 1.3.

While the neural network does a good job of face recognition in complex environments, the office world does not require this level of sophistication. Therefore, the tradeoffs in using the

algorithm in an office setting with controlled conditions are not sufficient to make it suitable for biometric network access.

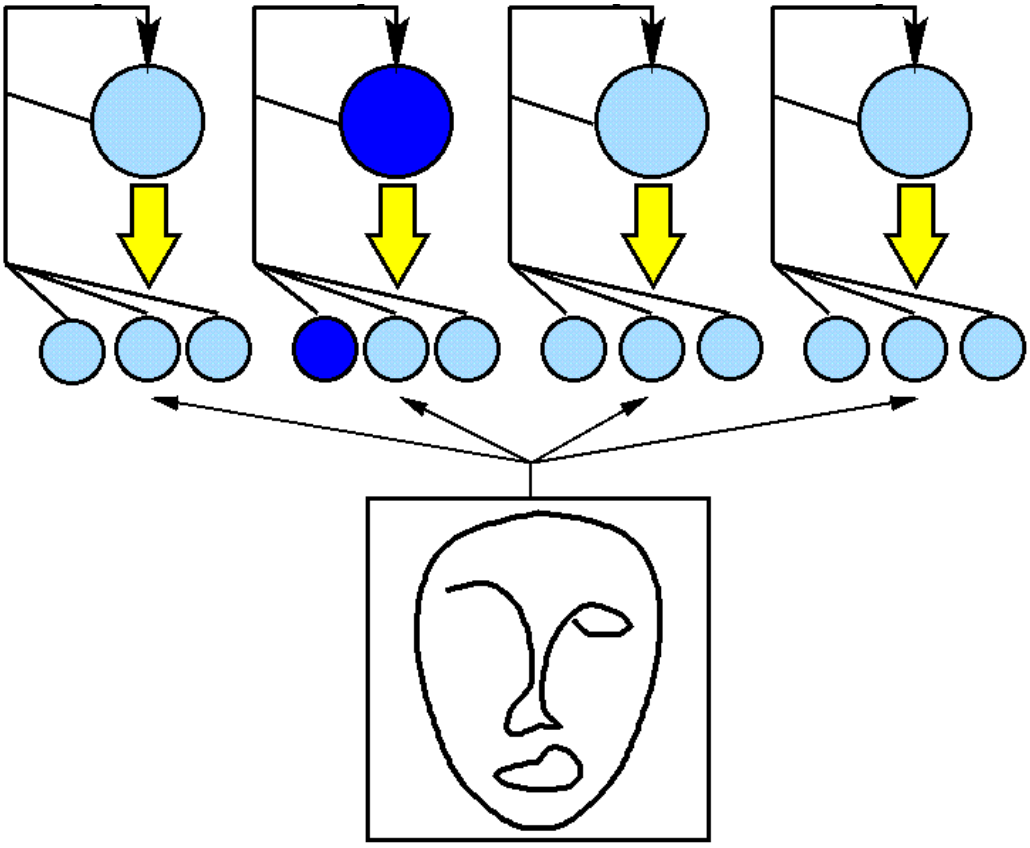

Figure 1.3: Neural network based face recognition

1.4.2 Iris Biometrics

The iris is the only internally visible organ of the human body. It is located in the eye behind the cornea and aqueous humor, and it is an ideal trait for measurement. It is protected by the eyelid and cornea, and is not exposed to harsh conditions that may cause it to be difficult to image. The iris, like the fingerprint, forms during the early stages of fetal development and is completed by the eighth month. It is extremely distinctive and will not be the same even for genetically identical twins.

The use of the iris for biometric authentication is relatively new [7][8]. All the current commercial algorithms are based on the original patented algorithm from John Daugman at the University of Cambridge. In 1994, the iris engine was ready and available for licensing. Since then, Iridian Technologies has purchased the algorithms and associated rights. Iridian Technologies has licensed other companies to build applications that leverage the iris algorithms.

In Iris biometrics, the iris image as shown in Figure 1.4 is acquired through iris scanner. Next iris is segmented from the background of the image. Then feature extraction algorithms are applied to extract the features from iris. Different types of feature extraction algorithms are available. Most widely used algorithm is based on Gabor filters. A set of Gabor filters were used to extract iris features. Another method based on Principal Component Analysis is also popular in iris recognition.

In iris biometrics, a strong, reliable biometric trait is measured, generating a template that is simple to compare and provides virtually no false accept rate (FAR). There is also an extremely low false reject rate (FRR) of 0.2% in three attempts. With a very high FAR and a very low FRR, iris biometrics work very well for both identification and verification. It is clear that the iris can deliver the best level of accuracy of all other biometrics.

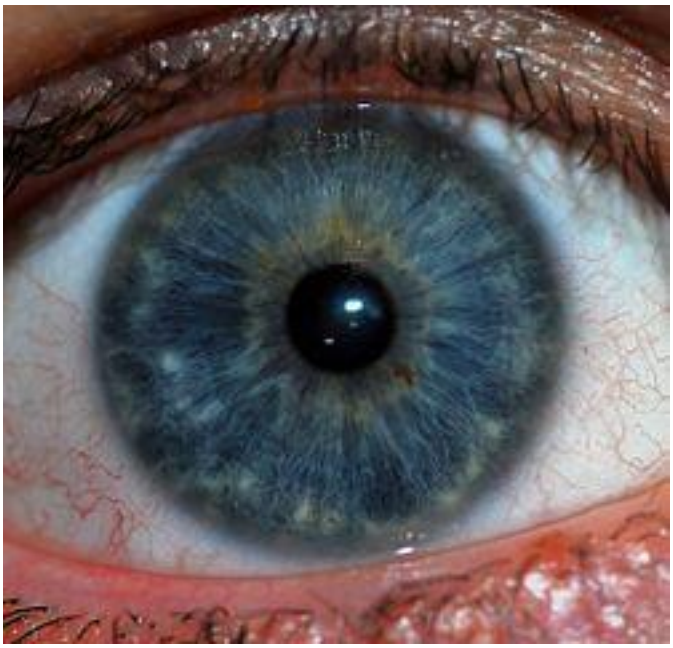

Figure 1.4: An Iris image

1.4.3 Fingerprint Biometrics

A fingerprint consists of ridges and valleys on the surface of a finger [9][10]. Ridges and valleys are often run in parallel and sometimes these are bifurcating and sometimes these are terminating. When a ridge comes to an end, it is called termination and when single ridge divides into two ridges, this is called bifurcation. A fingerprint image is shown in Figure 1.5.

Fingerprint based identification and verification is one of the most important biometric technologies which are now widely used in forensics and civilian applications which include border access control systems, machine readable passports, computerized identification cards, ATM banking security and information security systems which provide a unique ID to the citizens. The performance of an Automatic Fingerprint Identification System (AFIS) depends heavily on the feature extraction and matching algorithms. In biometrics community, fingerprint matching is an active research area. Researchers have explored different fingerprint feature extraction and matching algorithms to increase the matching performance. Different types of fingerprint matching algorithms and major issues in fingerprint matching are discussed in detail in chapter 2.

Figure 1.5: A Fingerprint Image

1.5 Thesis Objective

The main objective of this thesis is to study the fingerprint quality estimation and fingerprint matching techniques and propose new fingerprint quality estimation and matching framework in order to improve the matching performance of automatic fingerprint identification systems.

1.6 Research Findings

A novel k-means clustering based fingerprint quality estimation technique has been proposed which improves the performance of minutiae based fingerprint matching system. This technique classifies the fingerprint image as good, dry, wet or normal. For fingerprint matching, a novel technique based on Zernike moments has been proposed. This technique utilizes the Zernike moments as features and these moments are compared in matching stage. The Zernike moments are rotation invariant and this property is utilized to handle rotation in a fingerprint image. Experimental results show that Zernike moment based fingerprint matching technique outperforms the Gabor filter based matching techniques.

1.7 Thesis Outline

The rest of this thesis is organized as follows: Chapter 2 discusses the fingerprint identification algorithms and sets up the background for remaining chapters. Chapter 3 presents the k-means clustering based fingerprint quality estimation. Chapter 4 discusses the fingerprint matching algorithm based on Zernike moments. In chapter 5 experimental results of fingerprint matching are discussed. The dissertation ends with conclusions and future work discussed in chapter 6.

Chapter 2

Fingerprint Identification

2.1 Introduction

The process of identifying a person through their fingerprint is done by analyzing the pattern that exists on the underside of every finger. These patterns consist of composite curve segments and are unique for every finger, and for every person. The pattern contains both light shaded curves, which are referred to as the *ridges* of the fingerprint and dark shaded curves which are referred to as the *valleys* or furrows of the fingerprint as shown in Figure 2.1. Note that a fingerprint image is actually a reverse of the original print on a person's finger. That is, the ridges are represented as the dark-shaded areas, where the valleys are the light-shaded areas. The flow of these ridges and valleys form up the unique fingerprint pattern, and creating an automated process that matches fingerprints based on this pattern is the goal of fingerprint identification.

Although the notion of identifying a person through their fingerprint was practiced since the 16th century, the first scientific study of fingerprints was carried out in the late 19th century. The research that was carried out at this time formed the foundation for which modern fingerprint authentication systems operate. The research was primarily carried out by two individuals, E. Henry and F. Galton [11][12] towards the end of the nineteenth century. Each researcher investigated in different facets of fingerprint identification, and both of their work has provided a

valuable contribution to the area of fingerprint identification. Their study led to the formal acceptance of fingerprints as a valid means of identifying an individual

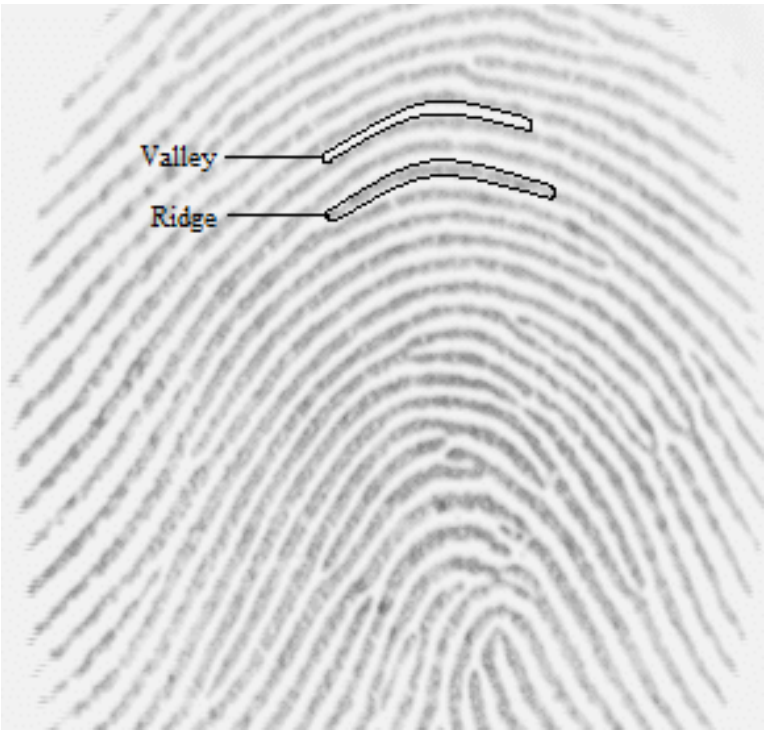

Figure 2.1: Fingerprint ridge and valley

2.2 Fingerprint Classification

The research carried out by Henry [13] dealt with the global or macro structures of fingerprint patterns. He analyzed many sets of prints, and his results from this analysis have produced a classification system for fingerprints, known as the *Henry System*. This classification scheme categorizes all fingerprints into 5 broad groups, which are based on the global structure of the fingerprint pattern. These categories are: Right Loop, Left Loop, Twin Loop, Whorl, Arch and Tented Arch as shown in Figure 2.2.

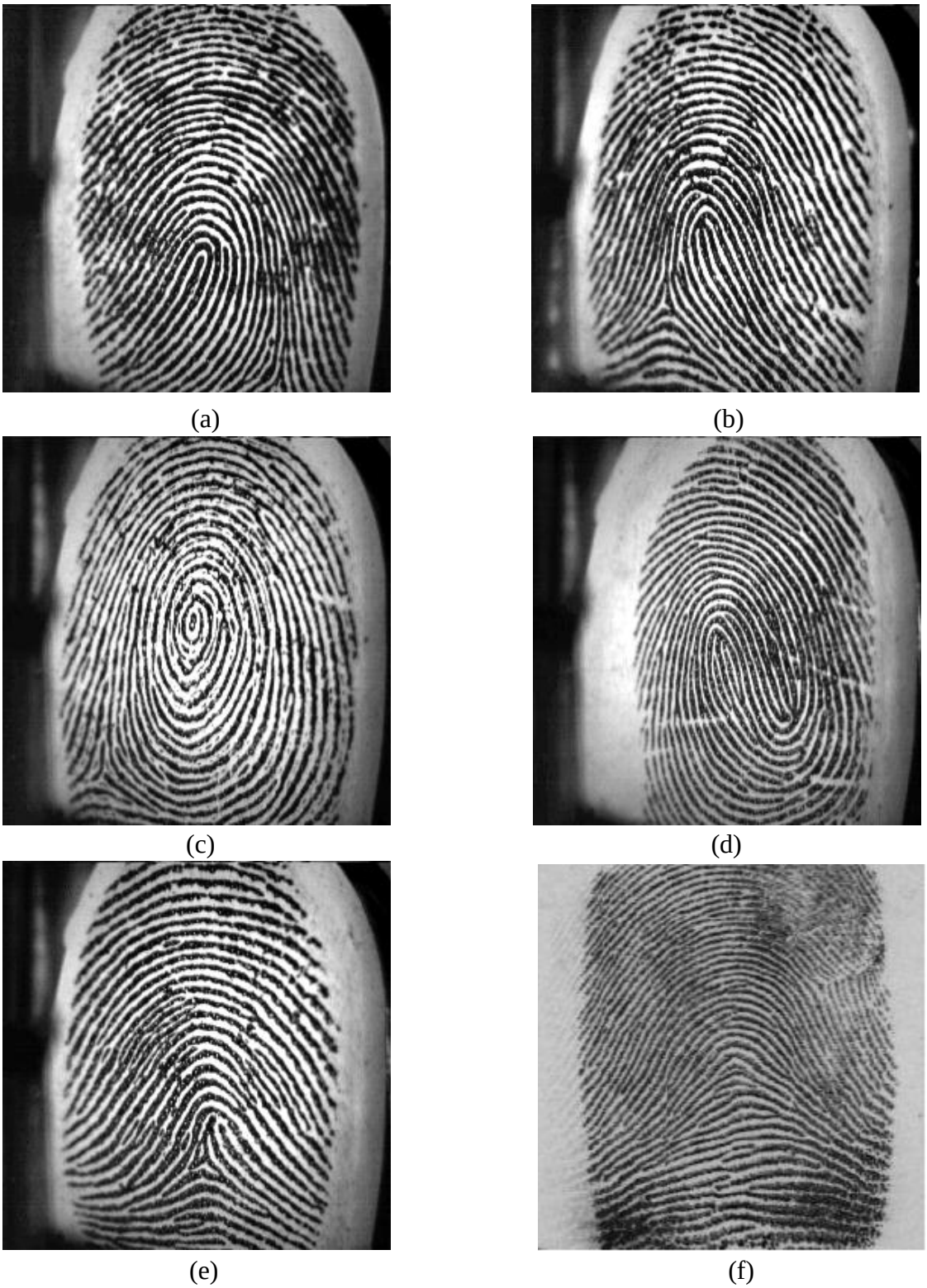

Figure 2.2: (a) Left Loop, (b) Right Loop, (c) Whorl, (d) Twin Loop, (e) Tented Arch, (f) Arch

The three basic ridge patterns found in fingerprints are the loop, arch, and whorls. Approximately 65% of all fingerprints have loop patterns, 30% arch patterns, 4% whorl patterns, and 1% have patterns other than the basic types [10][14][19].

2.3 Features and Uniqueness in Fingerprints

The research carried out by Galton's [11][12] dealt primarily with two separate topics. In one area of research, he investigated the uniqueness property of a fingerprint. Up until then, fingerprints were assumed to be relatively unique, but there was no certainty that the uniqueness property held. Galton's findings have now ensured that every fingerprint is in fact unique.

Galton also examined the changes in the fingerprint pattern structure over age. His results showed that a fingerprint pattern remains the same through age, and this finding together with the uniqueness property has allowed authentication through fingerprint recognition a possibility.

Galton's other area of research dealt with the micro structure of a fingerprint pattern. He revealed the existence of small discontinuities in the fingerprint ridge pattern flow, called *minutiae* points. Furthermore, he was able to classify these minutiae points into categories based on the type local discontinuity that existed. The significance of the finding was that these minutiae points could be used as reference points to aid in the authentication of a person through their fingerprint.

The original set of minutiae points, or Galton features consisted of four feature types. This set has now been extended to incorporate more minutiae point types, and is referred to as the Extended Galton Feature set. The contents of this set are: Dot, Ridge End, Lake or Enclosure, Bifurcation, Short Ridge, Crossover or Bridge, and Spur as shown in Figure 2.3

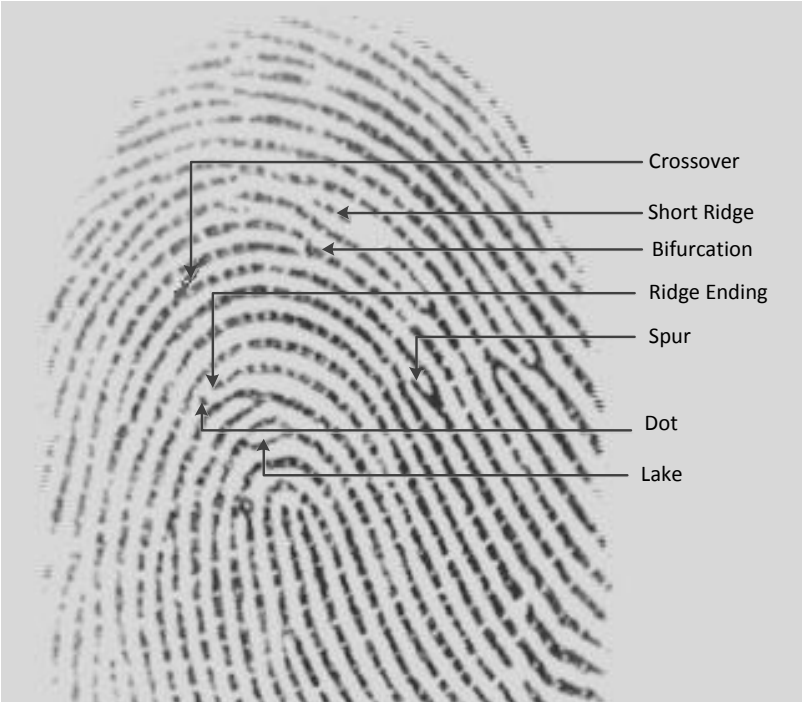

Figure 2.3: Extended Galton Feature Set

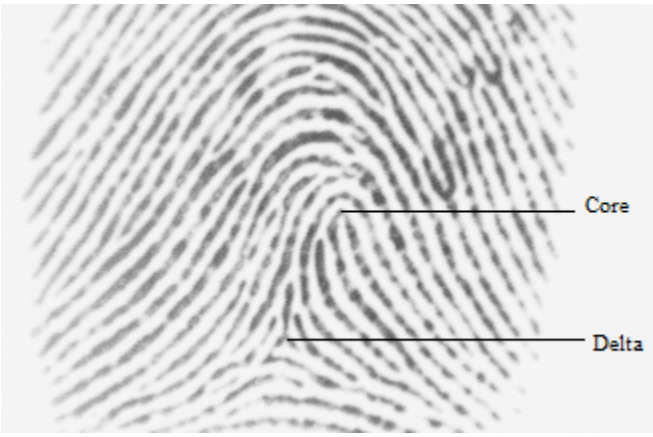

Figure 2.4: Core and Delta Points

Two other key features of a fingerprint are the core and delta points. These are also called as the singularity points of the fingerprint. The core point is defined as the top-most point of the innermost ridge and delta points are defined as the point in which three flow patterns meet. Figure 2.4 shows core and delta points of a fingerprint.

2.4 Fingerprint Matching

In fingerprint matching, first the fingerprint image is acquired through fingerprint scanner and then preprocessing is done on a fingerprint image. Next features are extracted from a fingerprint image and these features are stored as template in a database. When a query fingerprint image is presented to a fingerprint matching system, the features are extracted and compared with the stored templates of fingerprints in a database. Then based on some threshold values, it is decided that fingerprint image is matched or not. The query and template fingerprint image alignment is a major problem in a fingerprint matching. When acquired through fingerprint scanner, the query fingerprint image may be translated (displaced) left, right, up or down as compared to template image. Also the query fingerprint image may be rotated as compared to a template image. So before matching, translation and rotation issue of a query fingerprint image must be resolved. Both the query and template fingerprint images should be aligned with respect to each other and there should be no translation and rotation difference.

Some fingerprint matching algorithms operate directly on grayscale fingerprint image [15][17][18] while most others requires that an intermediate representation of fingerprint image is derived through feature extraction stage.

Matching of fingerprint images is a very difficult problem because there are large intra-class variations (translation and rotation in different impressions of same finger). The main reasons for these intra class variations [16][19] are given below:

- The same finger may be placed at different area of fingerprint scanner surface so the fingerprint image may be displaced left, right, up and down. Hence instead of capturing complete fingerprint image, only its partial left or right portion may be captured.

- The same finger may be rotated at different angles with respect to fingerprint scanner surface during different acquisitions of fingerprints.

- In fingerprint scanning, the three dimensional shape of finger is mapped on to the two dimensional scanning surface of the scanner. This mapping results in a non-linear distortion in different images of same finger due to the skin plasticity in different fingerprint acquisitions.

- The ridge and valley structure of a fingerprint would be captured accurately if the ridges and valleys of a fingerprint were in uniform contact with the scanner. Due to the difference in fingerprint pressure, skin dryness, grease and humidity in the air, the finger is not uniformly contacted with the scanner surface..

- There is a noise which is caused by surface of fingerprint scanner. The surface of scanner may get dirty after large number of fingerprint acquisitions. Therefore acquired fingerprint image is very noisy.

- The feature extraction algorithms are not 100 percent perfect and errors are introduced by these algorithms during different stages.

2.5 Fingerprint Matching Techniques

There are many fingerprint matching methods [18][20][21][22][23][24][25][26]. These fingerprint matching methods can be mainly classified into three categories.

- Minutiae based fingerprint matching
- Feature based fingerprint matching
- Correlation based fingerprint matching

2.5.1 Minutiae based fingerprint matching

This is the most popular technique in fingerprint matching. This technique is based on matching process used by human fingerprint examiners. Minutiae are extracted from the fingerprint image and stored as feature vectors. These feature vectors contains minutiae coordinates, their orientation, their angle with respect to other minutiae and their distance from other minutiae. The minutia based technique [20] first finds the minutiae in query fingerprint image and matches

their features [17] in a stored template fingerprint. There are 60 to 80 minutiae in a good quality fingerprint image, but there are different number of minutiae in different fingerprints.

The main steps of the minutiae-based matching system are following:

a. Estimate the direction field to establish the orientation of ridge-valley structures in the fingerprint.
b. Perform adaptive filtering to reduce noise.
c. Obtain a binary image of the fingerprint by performing thresholding.
d. Perform thinning to obtain ridges with a width of 1 pixel.
e. Extract minutiae points from the thinned image.
f. Remove false minutiae points from the thinned image.
g. Register minutiae templates in the biometrics database.
h. Match the fingerprints by comparing the freshly captured fingerprint with the registered minutiae templates.

2.5.2 Feature based fingerprint matching

In Minutiae based matching approaches, the poor quality fingerprint images are a major problem as minutiae points cannot be easily obtained from these poor quality fingerprints. There are other fingerprint features which includes ridge frequency, ridge shape, ridge orientation and texture information. These fingerprint features can be extracted more reliably than minutiae. In ridge feature based matching, fingerprints are compared in terms of their extracted features from ridge pattern. This technique is a feature-based technique that captures the ridge features from a

fingerprint image and stores them as a feature vector [15][21][22]. The fingerprint matching is based on the similarity between the two corresponding feature vectors and therefore the matching process is very fast. There are other feature based approaches in which features are Zernike features, moment features, frequency features and wavelet features. These features can easily be extracted from a fingerprint, and matching based on these features is very fast.

The main steps of the ridge feature based matching system are following:

a. Estimate the direction field to establish the orientation of ridge-valley structures in the fingerprint.
b. Perform filtering to reduce noise in direction field..
c. Find the reference point in fingerprint image.
d. Tessellate the image in circular sectors or in square grid.
e. Apply bank of Gabor filters to region of interest in fingerprint image.
f. Calculate variance in the filter image (feature vector).
g. Compute Euclidean distance between template image and query image.

2.5.3 Correlation based fingerprint matching

Correlation-based techniques rely on the broader and overall correlation between two images. Depending upon the correlation technique, two fingerprint images are aligned and a correlation between corresponding pixel values is computed. Correlation based techniques [23][24][25] matches the global patterns of holistic features of the fingerprints and therefore these are more tolerant to fingerprint image degradation.

To use the correlation-based verification system:

a. Obtain the characteristic information from the captured fingerprint image
b. Create a primary template of the fingerprint by performing image normalization
c. Find the characteristic positions in the primary template using correlation computation technique
d. Match the characteristic positions of the primary template with the secondary template stored in the biometrics database
e. Find the maximum correlation in both the primary and secondary template to decide whether the prints match
f. Obtain the correlation value to verify

2.6 Summary

In this chapter, different types of fingerprint classes based on singularities have been briefly discussed. Major issues in fingerprint matching have also been highlighted in this chapter which causes problems in matching. These issues must be resolved by a fingerprint matching algorithm to get the best results in matching. Different categories of fingerprint matching methods have also been explained in this chapter.

Chapter 3

Fingerprint Quality Classification

3.1 Introduction

Fingerprint identification is most popular biometrics used in the identity verification, criminal investigations and other commercial applications. Fingerprint identification is one of the leading biometrics used in criminal investigation, border access control systems, national identification cards, ATMs, airports and other security installations. Fingerprints are believed to be unique across individuals and across fingers of same individual. Even identical twins having similar DNA, are believed to have different fingerprints [27]. These observations have led to the increased use of automatic fingerprint-based identification in both civilian and law-enforcement applications.

A fingerprint is the pattern of ridges and furrows on the surface of a fingertip. Ridges and valleys are often run in parallel and sometimes these bifurcate and sometimes these terminate. These bifurcations and terminations are called minutiae. The performance of fingerprint identification system depends heavily on the quality of the fingerprint image [28][29]. If the fingerprint image quality is poor then there will be a more probability of error in fingerprint matching. Poor quality fingerprint image contains many spurious minutiae which degrades accuracy of the fingerprint matching system.

3.2 Fingerprint Quality

The quality of fingerprint image effects the accurate extraction of minutiae. The low quality image contains large number of false minutiae as compared to good quality image. Figure 3.1 shows the dry, good, wet and normal quality of fingerprint images. Fingerprint quality [29][30] is usually defined as a measure of the clarity of ridges and valleys. Good images have very clear ridges and valleys. Wet images have thicker dark ridges and thin valleys while the dry images have very thin light ridges. Normal images are those which have 60% to 70% clear ridges and 30% to 40% light ridges. Many factors affect the quality of the fingerprint image. The fingerprint scanner surface is dirty which degrades the quality of the acquired image. The surface of the fingerprint may also be dry. Person applies less pressure of the fingerprint on the scanner surface which also degrades the acquisition quality. So there are many factors which are related to the environment or user body which affects the quality of the fingerprint image. A fingerprint consists of ridges and valleys which run in parallel. In good quality fingerprint images, the ridges and valleys are clearly defined. But in low quality images, the ridge and valley structure is not well defined. Fingerprint quality is utilized to improve the quality of database and acceptability of fingerprint image during enrollment. The performance of fingerprint identification systems depends heavily on the quality of the fingerprint image. The performance degrades significantly when the input image is of low quality. The different conditions of the skin of the finger and limitations of the fingerprint scanners affect the quality of the acquired images.

Various techniques have been proposed in the literature for estimation of fingerprint quality. Hong et. al. [31] computed the frequency of the each block of fingerprint image and marked each

Figure 3.1: a) Dry fingerprint image b) Good fingerprint image c) Wet fingerprint image
d) Normal fingerprint image

block into recoverable and non-recoverable regions. Shen et. al. [32] used a bank of gabor filters on blocks of fingerprint image to determine the quality of that image. Chen et. al. [30] used the fingerprint quality factor and quality index in frequency domain and spatial domain to predict the quality of fingerprint image. Chen et. al. calculated the energy rings in frequency domain and block-wise coherence in spatial domain to estimate the quality of the fingerprint image.

A survey of Different fingerprint image quality estimation techniques and their comparison has been discussed by Fernando et. al. [76]. The behavior of different quality measures in different estimation approaches has been discussed by Fernando et. al. [76]. Also the verification performance was analyzed for low quality images.

A quality estimation technique based on power spectrum has been described in [77]. The method discussed in [77] extract the sinusoidal-shaped wave along the direction normal to the local ridge direction and then complete its Discrete Fourier Transform. Shen et. al. [32] proposed a method based on Gabor features. In this technique Gabor filters with different direction have been applied on the blocks of the fingerprint image and their responses are analyzed. Good quality blocks have larger responses as compare to low quality blocks.

3.2.1. Proposed Fingerprint Quality Classification

This chapter proposed a novel hierarchical k-means clustering method for quality based fingerprint classification [78]. A set of features has been extracted in frequency domain and spatial domain. Then these features are utilized by fingerprint quality classifier to estimate the quality of fingerprint image as dry, wet, good and normal. An objective method has also been

proposed for performance evaluation of fingerprint quality classification. In the following sections, this technique has been described in detail. Section 3.3 describes the extraction of quality features in frequency domain. Section 3.4 explains the extraction of statistical features in spatial domain. Section 3.5 describes the k-means clustering based fingerprint quality classification. Section 3.6 describes the experiments conducted to evaluate the performance of fingerprint quality classification.

3.3 Quality Features in Frequency Domain

Chen et. al. [30] proposed a fingerprint quality estimation framework based on Fourier and spatial features. Fourier features have been calculated by the approach proposed by Chen. Enhancements have been proposed in our technique which improves the performance of the quality estimation technique. The quality features in frequency domain are used to classify the fingerprint image as good or poor (dry) quality image.

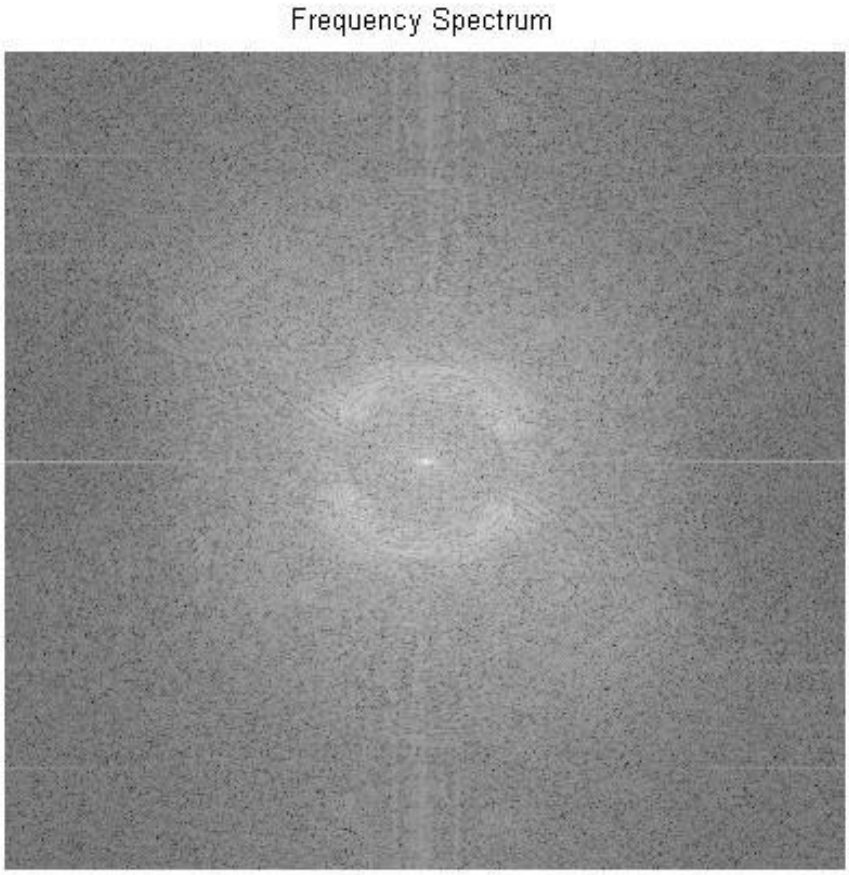

Figure 3.2: Spectrum of good quality fingerprint image

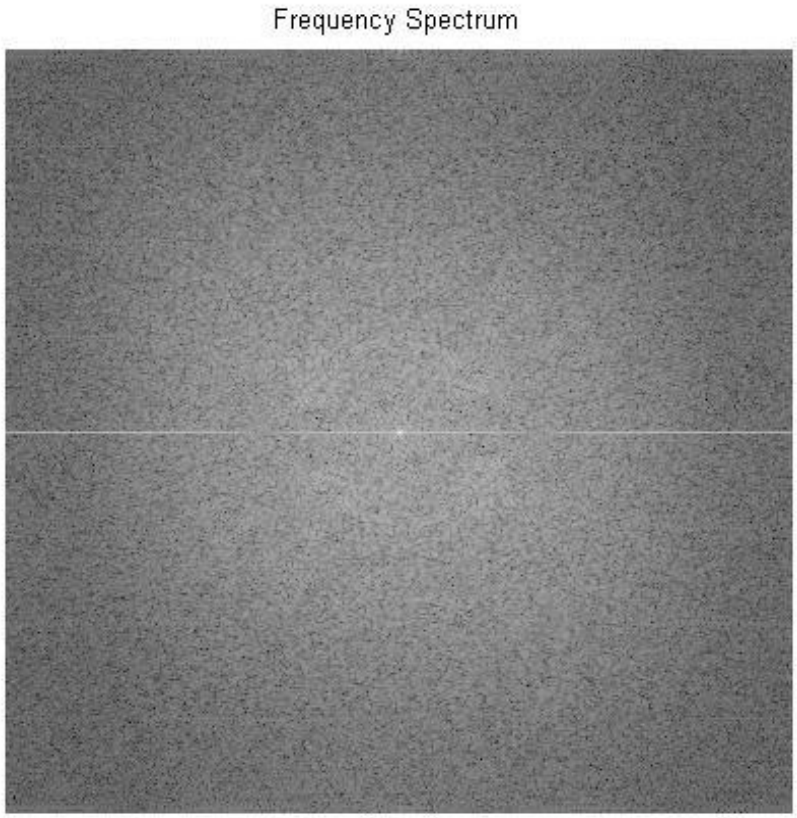

Figure 3.3: Spectrum of low quality fingerprint image

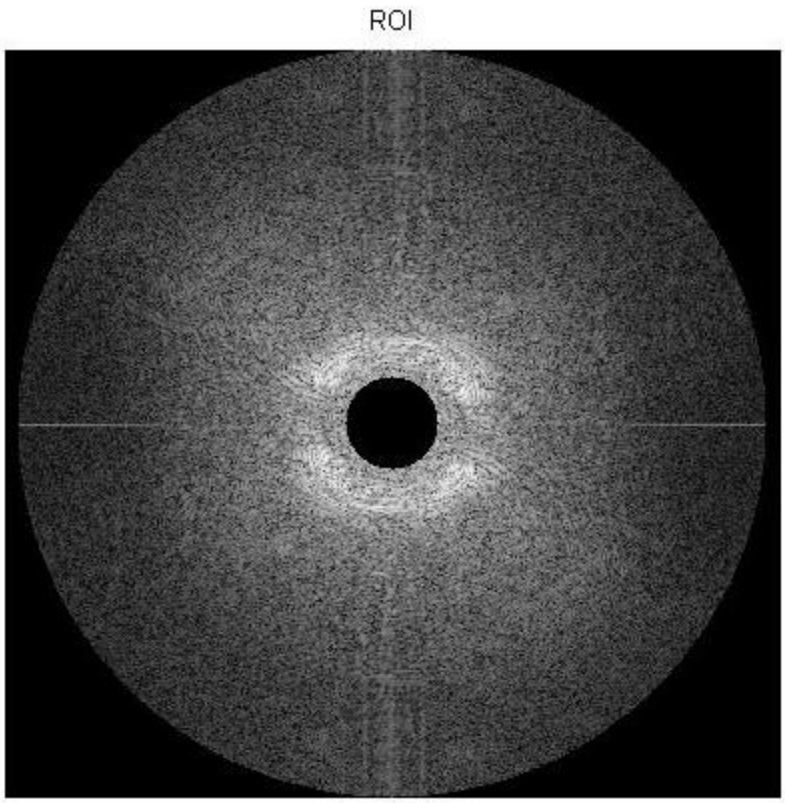

Figure 3.4: Region of Interest of good quality fingerprint image

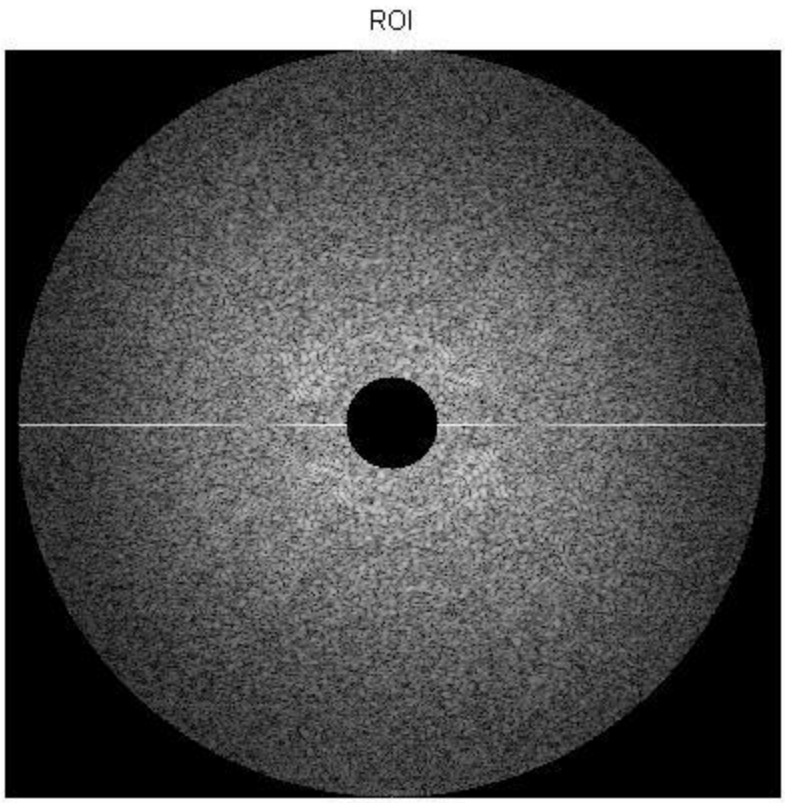

Figure 3.5: Region of Interest of low quality fingerprint image

The main steps of quality feature extraction algorithm in frequency domain are following:

i. The fingerprint image is transformed into frequency domain using Discrete Fourier Transform (DFT) [33]. The DFTs of a fingerprint image are shown in Figure 3.2 and Figure 3.3. Region of interest is shown in Figure 3.4 and Figure 3.5.

$$DFT = real\ (fftshift\ (fft2(image))) \qquad (3.1)$$

fft2 is 2 dimensional fast Fourier transform, *fftshift* is used to shift the minimum frequency at centre, real is used to take only real values from Fourier transform.

ii. The power spectrum [34] is obtained which is the square of the absolute of DFT of the image.

$$PowerSpectrum: P(k,l) = (abs(DFT))^2 \qquad (3.2)$$

where (*k,l*) is the pixel index in the power spectrum.

iii. Fifteen Butterworth band pass filters [30] are used to extract the energies in the annular bands from the power spectrum of the DFT. These filters are shown in Figure 3.6 and Figure 3.7.

Butterworth Low Pass filter:

$$H(k,l \mid m,n) = \frac{1}{1 + \frac{1}{n \times m^2}\left(\left(\frac{k-a}{M}\right)^2 + \left(\frac{l-b}{N}\right)^2\right)^n} \qquad (3.3)$$

Where *M* and *N* is the size of the image, (*k,l*) is the pixel index in the power spectrum corresponding to the spatial frequency $(\frac{2\pi k}{M}, \frac{2\pi l}{N})$ and (*a,b*) is the

central location of power spectrum corresponding to spatial frequency (0,0). The Butterworth function generates a low-pass filter with the cutoff frequency given by m & the filter order given by n. We construct a total of T=15 equally spaced Butterworth band pass filters, R_t, by taking difference of two consecutive Butterworth low pass filters. Equation for calculation of Butterworth band pass filters is given below.

Butterworth Band Pass filter:

$$R_t(k,l) = H(k,l \mid m_{t+1}, n) - H(k,l \mid m_t, n) \qquad (3.4)$$

iv. The energies in the t-th annular bands are extracted after convolution with band-pass Butterworth filters and these energies are denoted by E_t where t varies from 1 to 15 (T=15). E_t is calculated using equation (3.5).

$$E_t = \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} R_t(k,l) P(k,l) \qquad (3.5)$$

v. The normalized energy for the t-th band-pass filter is defined in equation (3.6).

$$P_t = \frac{E_t}{\sum_{t=0}^{T-1} E_t} \qquad (3.6)$$

vi. The extent of energy concentration is given by equation (3.7).

$$E = -\sum_{t=0}^{T-1} P_t \log P_t \qquad (3.7)$$

vii. Quality feature using all the 15 filters is calculated using equation (3.8).

$$Q_f = \log T - E \qquad (3.8)$$

viii. Quality feature $Q_{f\_limited}$ is also calculated for different number of filters. This calculation is done using the following equations.

$$P_{t\_limit} = \frac{E_t}{\sum_{t=start\_ring}^{end\_ring} E_t} \quad (3.9)$$

$$E_{limit} = -\sum_{t=0}^{T-1} P_{t\_limit} \log P_{t\_limit} \quad (3.10)$$

$$Q_{f\_limited} = \log T - E_{limit} \quad (3.11)$$

Start_ring is set to 1 whereas end_ring varies from 4 to 8. This corresponds to band-pass filters from 1 to 4, 1 to 5, 1 to 6, 1 to 7 and 1 to 8 which were used to extract energy from annular bands of power spectrum of DFT. Extracted ring energies of 15 filters are shown in Figure 3.8. This figure shows the ring energies in good and dry fingerprint images. Maximum amount of energy is concentrated in first 5 filters. Different ranges of filters have been tested and it is found out that quality feature calculated for filter 1 to 5 provides the best accuracy and minimum error in classification of fingerprint images as good or poor (dry) quality image. These classification results with different ranges of filters are shown in Figures 3.9, 3.10, 3.11,3.12.3.13 and 3.14. Therefore two quality features have been calculated Q_f and $Q_{f\_limited}$. All the extracted ring energies of 15 band-pass filters are used to calculate quality feature Q_f whereas the $Q_{f\_limited}$ is calculated using the extracted ring energies of first 5 band-pass filters.

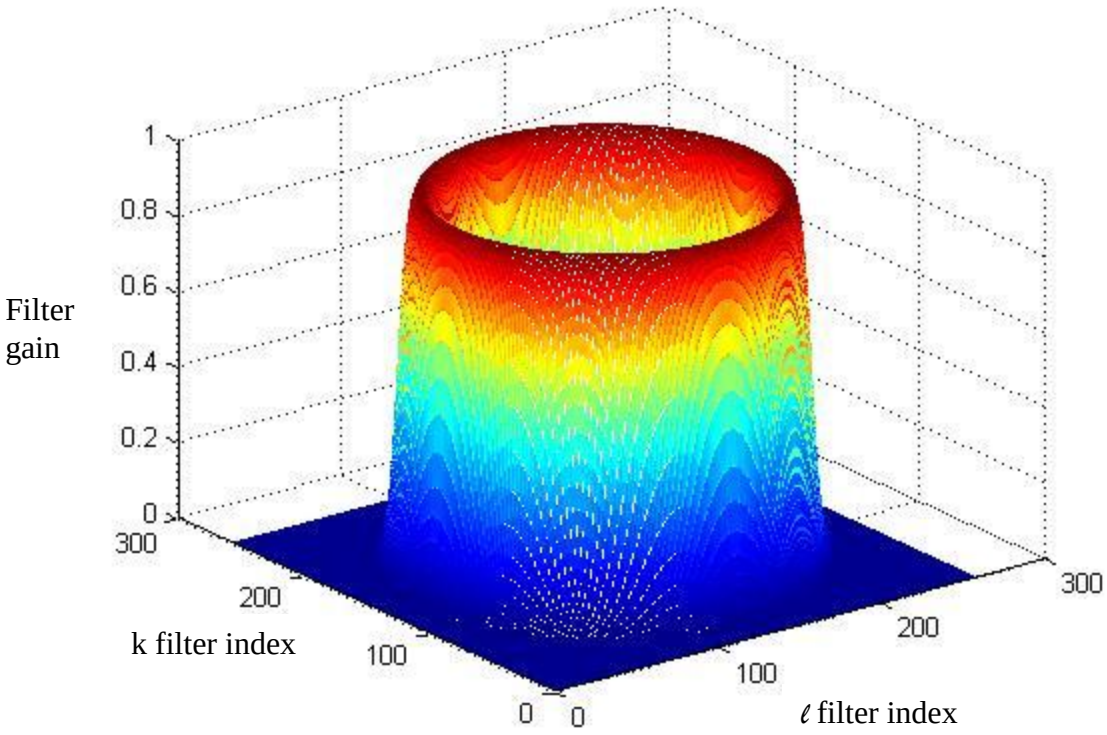

Figure 3.6: 3-Dimensional Butterworth band-pass filter

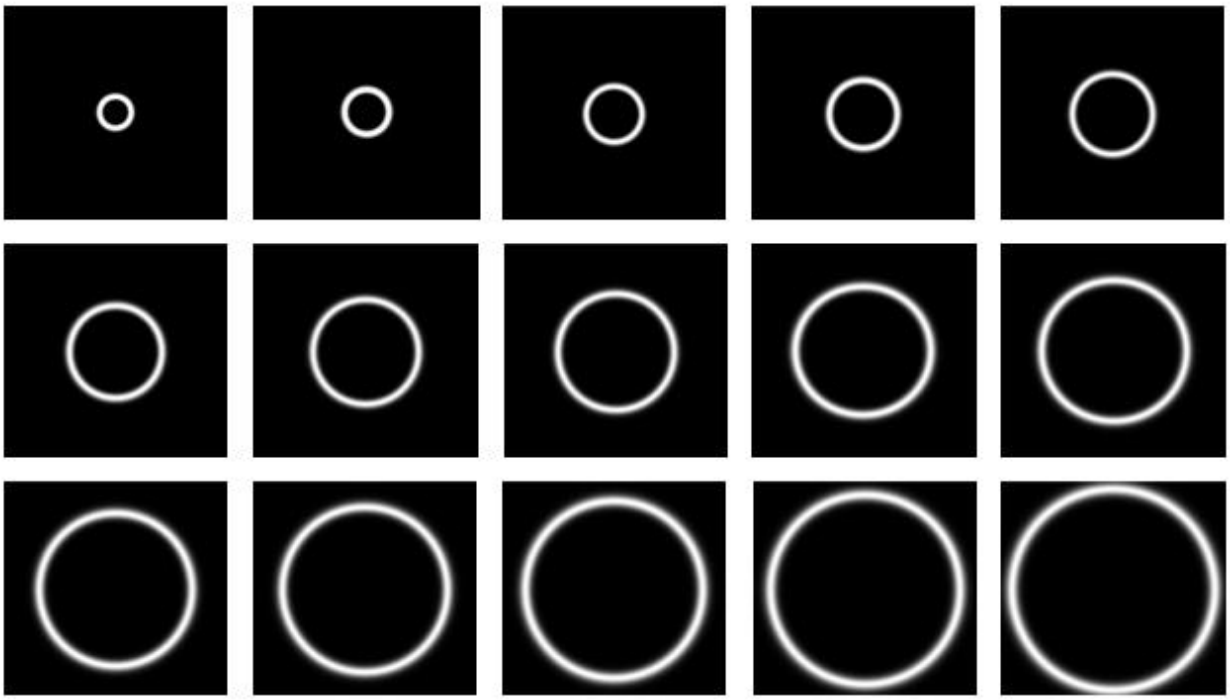

Figure 3.7: 2-Dimensional fifteen Butterworth band-pass filters. Top left is filter 1 and bottom right is filter 15.

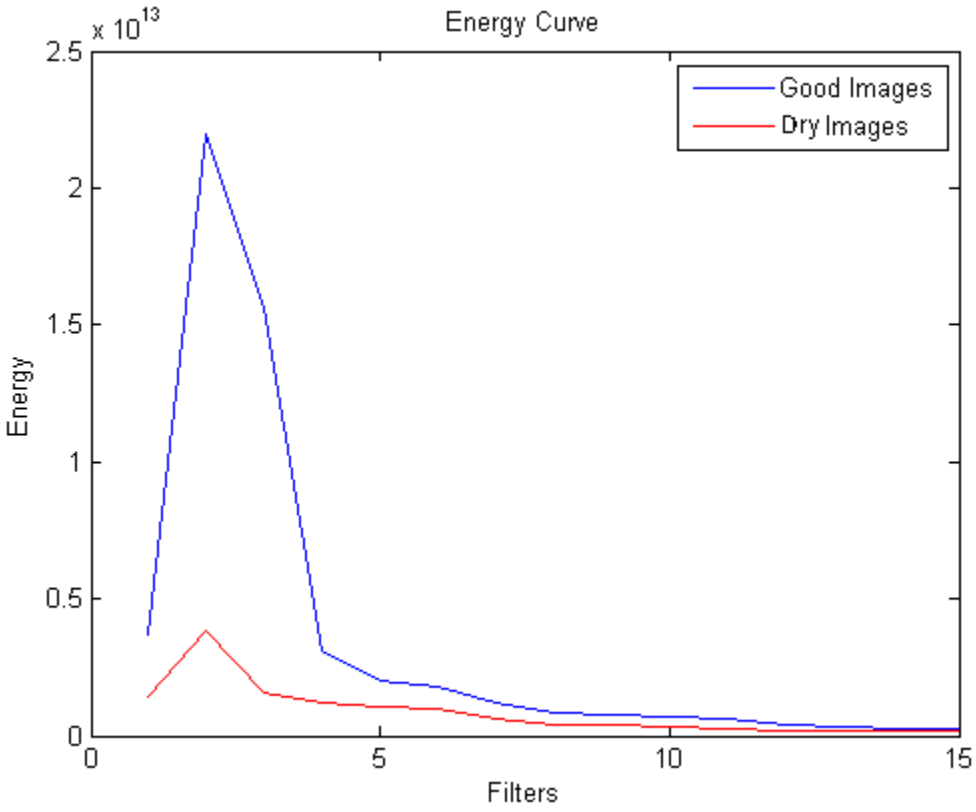

Figure 3.8: Plot of the ring energies from filter 1 to filter 15

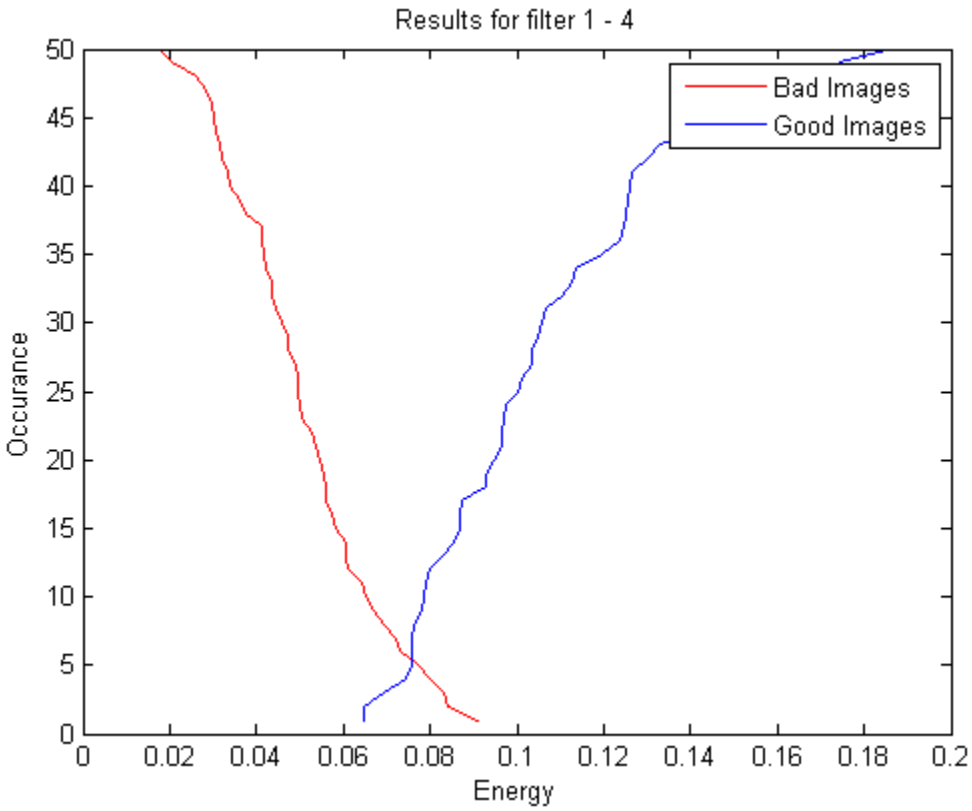

Figure 3.9: Results of classification of good and bad images when first 4 filters were used

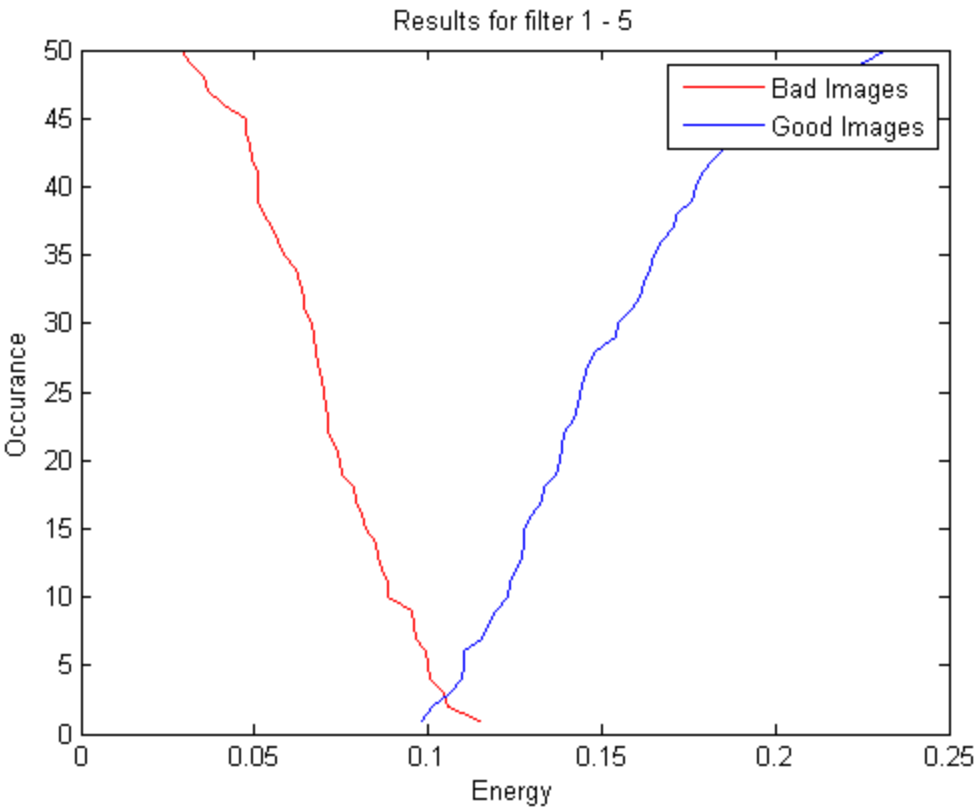

Figure 3.10: Results of classification of good and bad images when first 5 filters were used

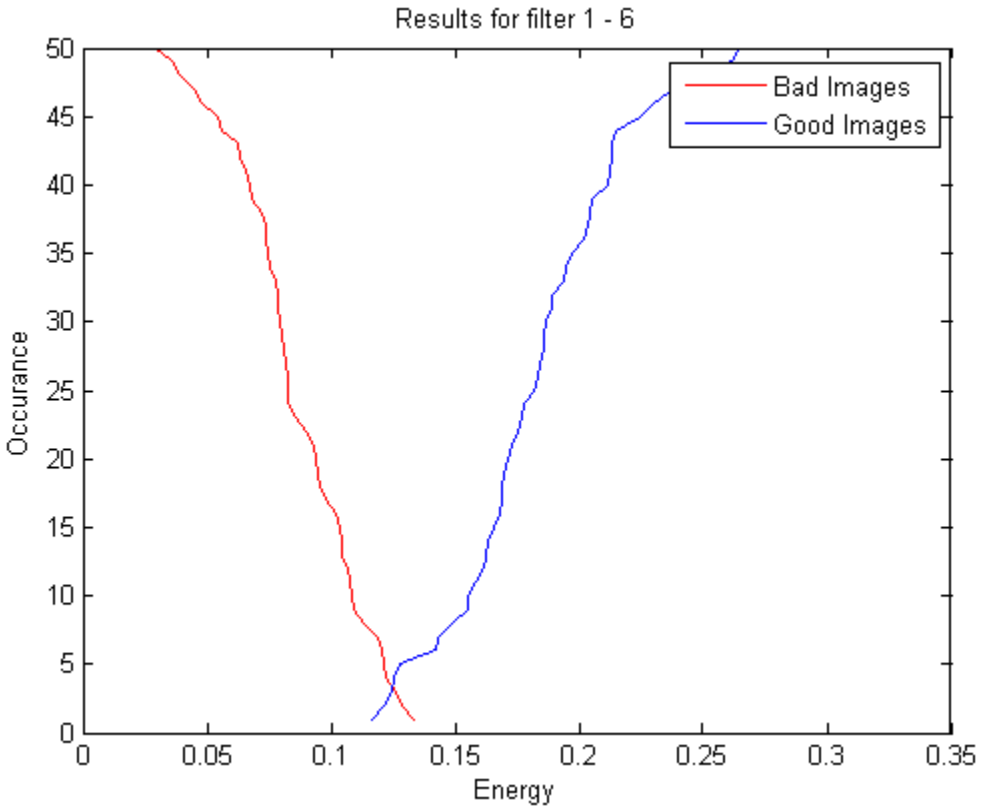

Figure 3.11: Results of classification of good and bad images when first 6 filters were used

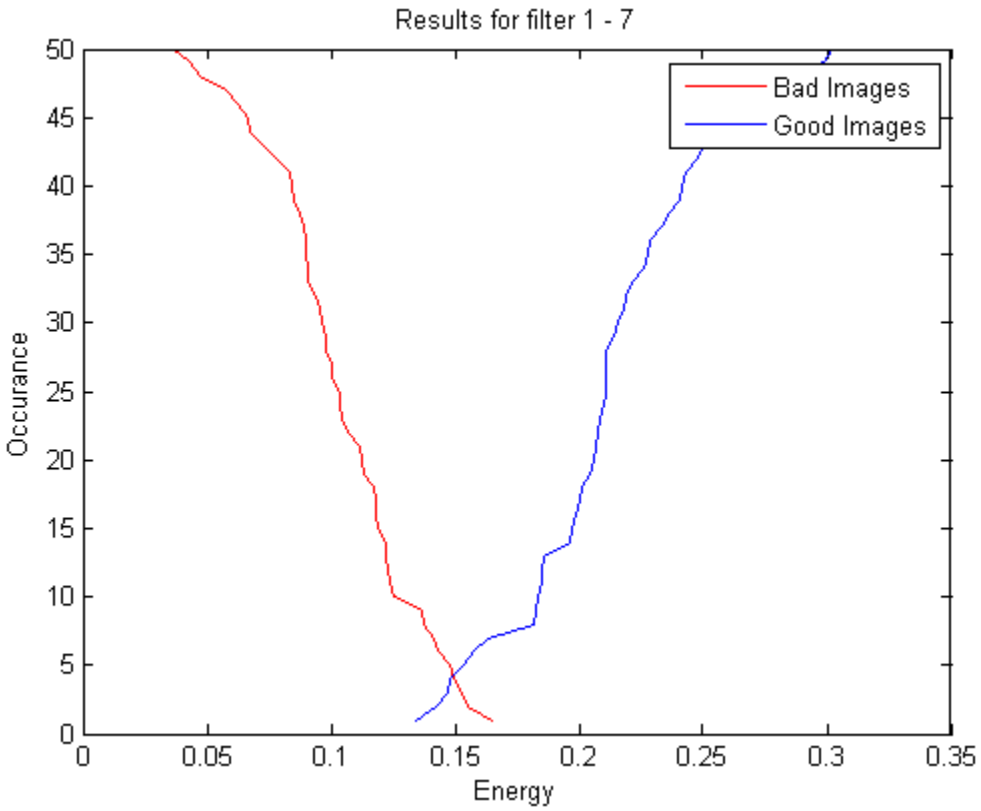

Figure 3.12: Results of classification of good and bad images when first 7 filters were used

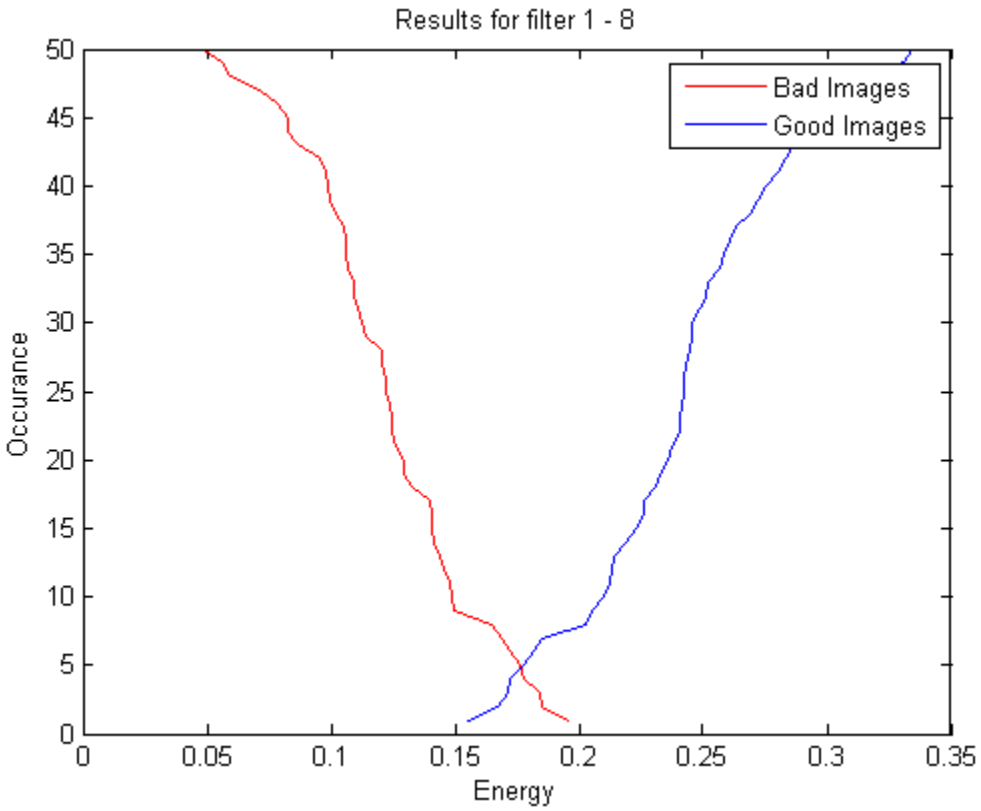

Figure 3.13: Results of classification of good and bad images when first 8 filters were used

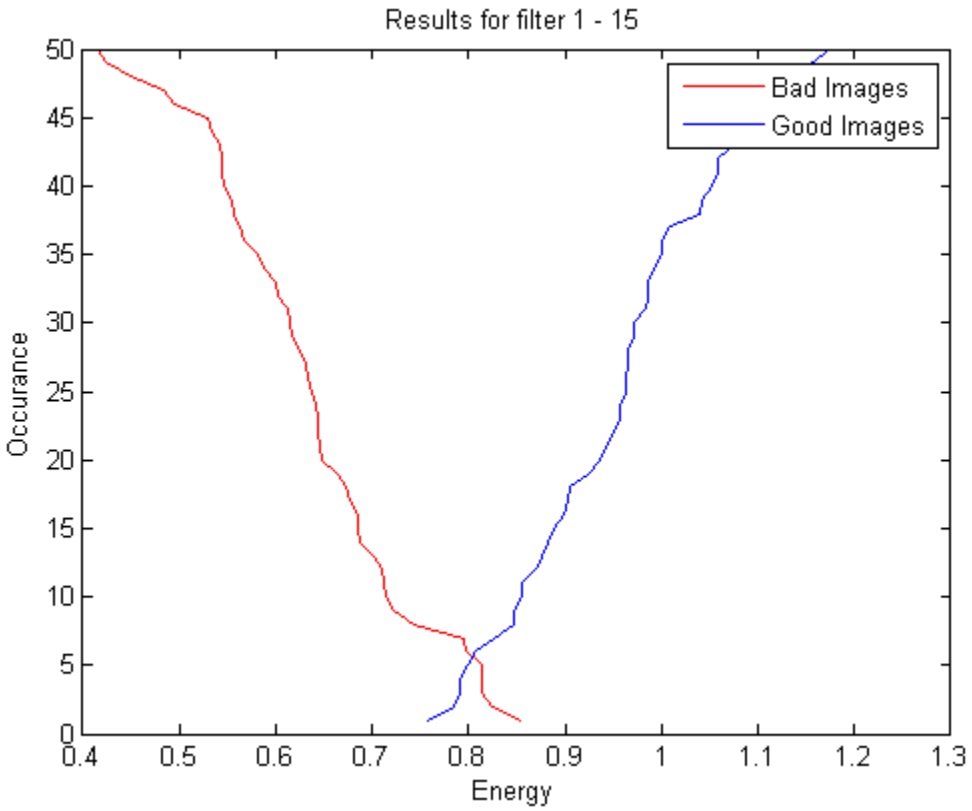

Figure 3.14: Results of classification of good and bad images when all 15 filters were used

3.4 Quality Features in Spatial Domain

In spatial domain, the foreground of fingerprint image was segmented from the background. The segmented fingerprint image consists of ridge and valley area of fingerprint. The segmented fingerprint image is divided into non-overlapping blocks of size 13x13. The 13x13 block size is chosen after experimentation with different block sizes and this 13x13 block size provides the best accuracy in segmentation of a fingerprint image. Statistical features have been calculated for each block of segmented fingerprint image.

3.4.1 Segmentation

Variance based segmentation algorithm has been implemented. Segmentation algorithm has the following steps:

1. Apply edge detection filter (i.e. canny) [35] on a fingerprint image.
2. Divide the fingerprint image into non-overlapping blocks of size 13x13.
3. Calculate variance of each block as given in equation (3.7).

   $$\text{Variance} = \frac{1}{n}\sum_{j=1}^{n}(x_j - M)^2 \qquad (3.7)$$

   M is the mean of pixel values of a block, x_j is a pixel value and n is the total number of pixels in a block.
4. A threshold value based on variance has been empirically determined. The blocks having fingerprint ridges and valleys have a larger variance value whereas blocks having no ridges and valleys have a smaller variance value.

Figure 3.15: Original image

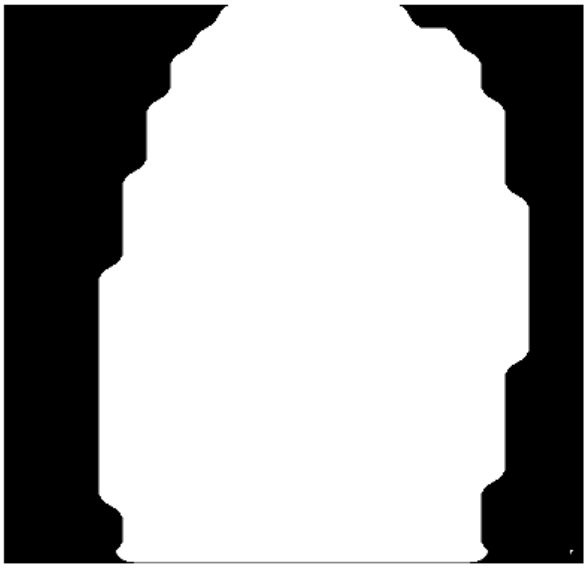

Figure 3.16: Segmented Mask

5. Based on this threshold value, the foreground of fingerprint image is segmented from the plain background. The original fingerprint image and its segmented mask is shown in Figure 3.15 and 3.16.

3.4.2 Statistical Features

We have calculated the statistical features [36][37] on the segmented fingerprint image. The segmented image is divided into non-overlapping blocks of size 13x13. As the average ridge width is found to be approximately 8~9 pixels in a fingerprint image, therefore a 13x13 block size is chosen which captures a ridge or part of a ridge. This 13x13 block size also provides best accuracy in the segmentation of a fingerprint image. Statistical features of each block have been calculated using the following equations.

$$\text{Mean}(M) = \frac{1}{n}\sum_{j=1}^{n} x_j \quad (3.8)$$

$$\text{Standard Deviation } (SD) = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(x_j - M)^2} \quad (3.9)$$

(x_j is the jth pixel value and n is total number of pixels in a block)

$$\text{Uniformity } (U) = \sum Prob(Hist)^2 \quad (3.10)$$

$$\text{Smoothness}(R) = 1 - \frac{1}{1+SD^2} \quad (3.11)$$

$$\text{Image Inhomogeneity}(InH) = \frac{m*U}{SD*R} \quad (3.12)$$

$$\text{Product of mean and standard deviation } (PMnSD) = M * SD \quad (3.13)$$

$$\text{Ratio of mean and standard deviation } (MS) = \frac{M}{SD} \quad (3.14)$$

Mean of each feature value has been calculated by the following equations.

$$\text{meanSD} = \frac{1}{n}\sum_{j=1}^{n} SD \quad (3.15)$$

$$\text{meanMS} = \frac{1}{n}\sum_{j=1}^{n} \text{MS} \quad (3.16)$$

$$\text{meanM} = \frac{1}{n}\sum_{j=1}^{n} \text{mean} \quad (3.17)$$

$$\text{meanPMnSD} = \frac{1}{n}\sum_{j=1}^{n} \text{PMnSD} \quad (3.18)$$

$$\text{meanInH} = \frac{1}{n}\sum_{j=1}^{n} \text{InH} \quad (3.19)$$

n is total number of blocks in a segmented fingerprint image. Another feature globalMS has also been calculated using equation (3.20).

$$\text{globalMS} = \frac{meanM}{meanSD} \quad (3.20)$$

3.5 Fingerprint Quality Classification

The fingerprint images are divided into four classes: Good, Wet, Dry and Normal. The training database consists of 20 images of each class. There are total of 80 fingerprint images in training database. The hierarchical k-means clustering algorithm [38][39] has been used for quality classification of fingerprint images. There are 3 levels in this classification. In each level, fingerprint images are classified into two classes. First level classifies the fingerprint image into dry class or class of good, wet, normal images. Second level classifies the fingerprint image into wet class or class of good, normal images. The third level classifies the fingerprint image into good or normal class. The flowchart of fingerprint quality classifier is shown in Figure 3.17 and description of different classification levels shown in flowchart is given below.

3.5.1 Level-1 Classification

In level-1 classification, k-means algorithm is used to classify the fingerprint images into two clusters. Cluster 1 consists of dry images whereas cluster 2 represents the remaining images i.e.

wet, good and normal. Different set of features have been empirically tested to find the best feature set for classification into two clusters with minimum error. The best feature set contains the five features which are meanSD, meanInH, meanMS, Q_f and globalMS. In training phase, centroid values of both clusters have been calculated using k-means clustering algorithm. In testing phase, absolute difference 1 and 2 has been calculated between the query feature set and both centroid values. If difference 1 is less than difference 2 then quality of fingerprint image is dry and if difference 2 is less than difference 1 than fingerprint belongs to cluster 2.

3.5.2 Level-2 Classification

The dry fingerprint image has already been classified in level-1. The remaining fingerprint images are of wet, good and normal quality. In level-2 classification, fingerprint images are again classified into two clusters. Cluster 1 consists of wet fingerprint images and cluster 2 consists of remaining images i.e. good and normal. Different set of features have been empirically tested to find the best feature set for classification into two clusters with minimum error. Best feature set used in level-2 classification consists of meanSD, meanPMnSD, meanMS, meanM and globalMS. Centroid values of both clusters have been calculated in the training phase using k-means clustering algorithm. In testing phase, two square Euclidean distance based differences 1 and 2 have been calculated between the query feature set and both centroid values. If difference 1 is less than difference 2 then quality of fingerprint image is wet and if difference 2 is less than difference 1 than fingerprint belongs to second cluster of good and normal fingerprint images.

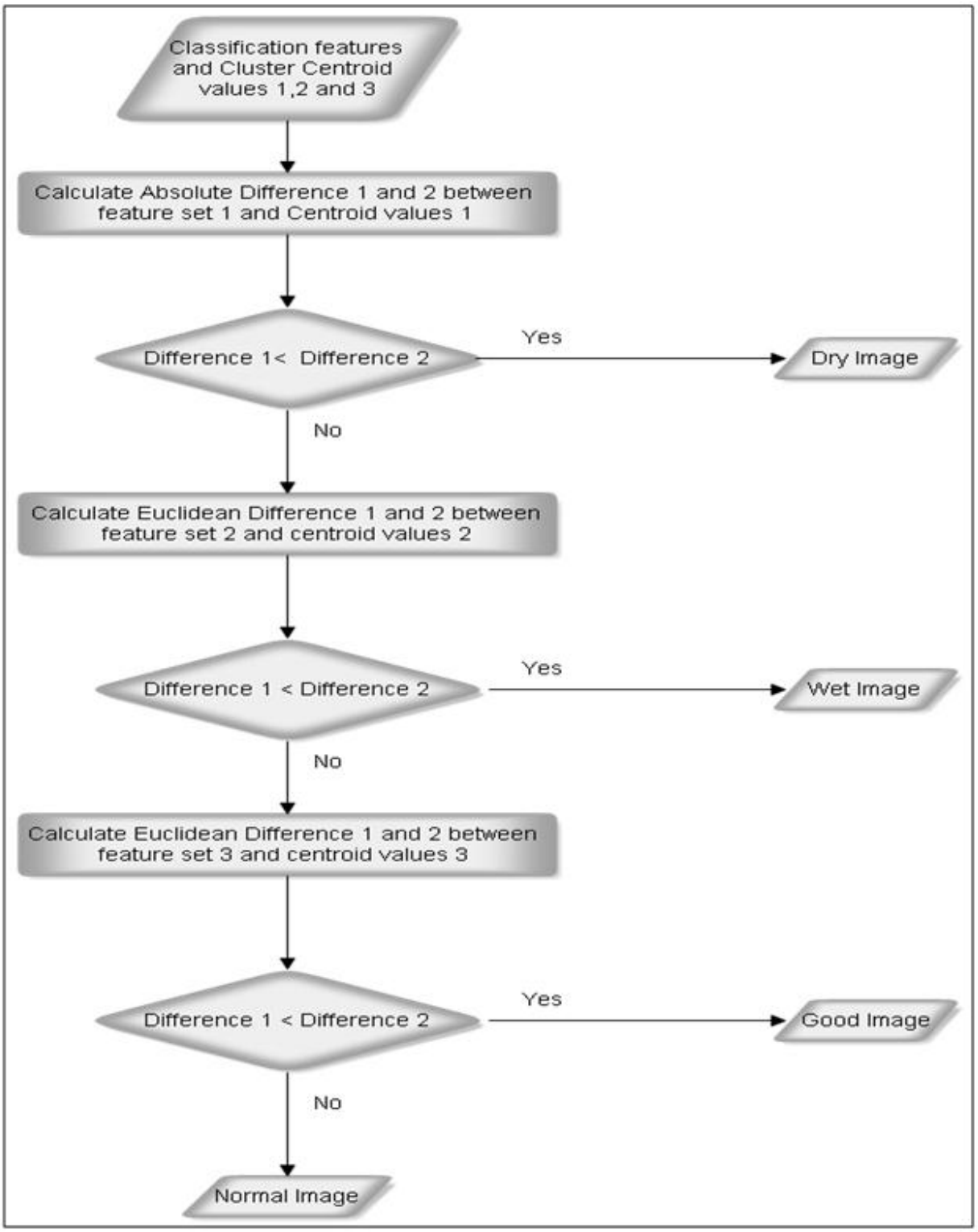

Figure 3.17: Flow chart of fingerprint quality classifier

3.5.3 Level-3 Classification

Now the dry and wet fingerprint images have already been classified. The good and normal quality fingerprint images are left for classification. In level-3 classification, fingerprint images are classified into good and normal clusters using k-means clustering algorithm. Different set of features have been empirically tested to find the best feature set for classification into two clusters with minimum error. Best feature set used in level-3 classification consists of meanSD, meanInH, meanM, Q_f and $Q_{f\_limited}$. In training phase, centroid values of both clusters have been calculated. In testing phase, two square Euclidean distance based differences 1 and 2 have been calculated between the query feature set and both centroid values. If difference 1 is less than difference 2 then quality of fingerprint image is good and if difference 2 is less than difference 1 than fingerprint image is of normal quality.

3.6 Experimental Results and Analysis

The experiments were performed on Intel Core 2 Duo 2.4 GHz processor running Microsoft Windows XP having 1 GB of RAM. The algorithms were implemented in MATLAB 2008. The quality classification results of training phase are shown in Table 3.1. Dry and normal quality fingerprint images are 100 % correctly classified in training phase whereas good and wet quality fingerprint images are 90% correctly classified in the training phase. The quality classification results of proposed approach tested on complete 800 fingerprint images of FVC 2002 db1 (Fingerprint Verification Competition 2002 database 1) are shown in Table 3.2. The classification error in each quality class is also shown in Table 3.2. An objective method has been proposed for performance evaluation of fingerprint quality classification. To test the efficacy of the fingerprint quality classifier, it has been incorporated in the minutiae based fingerprint matching system [40]. In minutiae based fingerprint matching method, total number

of genuine matches performed was 2800. There are 100 classes in FVC 2002 db1 database and each class contains 8 impressions of same finger. In each class, first impression is matched with all the remaining seven impressions of same finger, second impression is matched with all the remaining six impressions of same finger and so on. There are total 28 (7+6+5+4+3+2+1) genuine matches in each class. For 100 classes of FVC 2002 db1, there are total of 28 x 100 = 2800 genuine matches. The total number of imposter matches performed was 4950. In imposter matching, first fingerprint impression in class 1 is matched with the first fingerprint impression of all the remaining 99 classes. Then the first impression of class 2 is matched with the first impression of all the remaining 98 classes and so on. There are in total of 4950 (99+98+97.....+3+2+1) imposter matches.

We have divided the fingerprint images into 4 bins as shown in Table 3.3. In each bin, first is the query fingerprint image and second is the stored template fingerprint image. The Good-Good in bin 1 means the query image is good and template is good. The quality of the stored template of fingerprint image is already estimated and stored with the template. When the query fingerprint image is acquired by the fingerprint matching system, its quality is estimated at that time. Then based on the quality of query and template fingerprint image, these are allocated to one of the four bins. The number of genuine and imposter matches in each quality bin are shown in Table 3.4.

The matching score is normalized between 0 and 1. If matching score is 0, it means no match and if score is 1, it means 100 percent match. If the matching score is less than threshold then fingerprint is not matched and if score is greater than threshold then the fingerprint is matched.

For each quality bin, a threshold value for matching is varied from 0.15 to 0.5 and number of images which were falsely accepted and falsely rejected is calculated. The total number of false rejected images at different thresholds from all the 4 bins is calculated. Also the total number of false accepted images at different thresholds from all the 4 bins is calculated. The best threshold value for matching for each bin is empirically determined by experimentation. After performing experiments, it has been observed that at certain threshold values, the total number of false rejected images from all the 4 bins is 50 which is nearly same as in case of minutiae based fingerprint matching method without utilizing quality information. But the total number of false accepted images from all the 4 bins reduced to 39 while the total number of false accepted images in minutiae based fingerprint matcher is 90 without utilizing quality information. These results are presented in Table 3.5.

Image Quality	Total images	Correct Classification	Error
Dry	20	20	0%
Wet	20	18	10%
Good	20	18	10%
Normal	20	20	0%

Table 3.1: Classification results in training phase

Image Quality	Total images	Images classified correctly	Error
Dry	141	139	1.42 %
Wet	171	165	3.51 %
Good	172	164	4.65 %
Normal	316	300	5.06 %

Table 3.2: Results of proposed quality classification method tested on 800 images of FVC 2002 db1 database

Bin1 (Query-Template)	Bin 2 (Query-Template)	Bin 3 (Query-Template)	Bin 4 (Query-Template)
Good-Good	Good-Wet	Good-Dry	Wet-Wet
Good-Normal	Wet-Good	Dry-Good	Dry-Dry
Normal-Good	Normal-Wet	Normal-Dry	Wet-Dry
Normal-Normal	Wet-Normal	Dry-Normal	Dry-Wet

Table 3.3: Bins according to the quality of the query fingerprint and template fingerprint

	Bin 1	Bin 2	Bin 3	Bin 4	Total
Number of Genuine Matches	1074	627	557	542	2800
Number of Imposter Matches	1128	1776	720	1326	4950

Table 3.4: Number of Genuine and Imposter matches in each quality bin

	Bin 1	Bin 2	Bin 3	Bin 4
Threshold values of fingerprint matcher with quality classification	0.41	0.38	0.14	0.153
Number of false accepted images at these thresholds	4	5	24	6
Number of false rejected images at these thresholds	16	13	8	13
False Accept Rate (with quality classification)	(39/4950) x 100 = 0.79			
False Reject Rate (with quality classification)	(50/2800) x 100 = 1.8			
Threshold value of fingerprint matcher without quality classification	0.28			
False Accept Rate (without quality classification)	(90/4950) x 100 = 1.8			
False Reject Rate (without quality classification)	(51/2800) x 100 = 1.8			

Table 3.5: FAR and FRR of fingerprint matcher with and without quality classification

In a fingerprint matching system, there are four possible outcomes:

a. Genuine Acceptance
b. Imposter Rejection
c. Genuine Rejection (False Rejection)
d. Imposter Acceptance (False Acceptance)

The outcomes defined in (a) and (b) are correct while the outcomes defined in (c) and (d) are errors. The performance of a fingerprint matching system is given in terms of false accept rate (FAR) and false reject rate (FRR) [41][42][43][46]. The false accept rate and false reject rate of minutiae based fingerprint matcher is 1.8 without utilizing fingerprint quality classification when tested on fingerprint database FVC 2002 db1 [44][45]. When quality classification of fingerprint image is incorporated in the fingerprint matcher, the false accept rate has been reduced from 1.8 to 0.79 whereas the false reject rate is at 1.8. This show significant improvement in the reduction of false accept rate and efficacy of our proposed fingerprint quality classification method.

Chapter 4

Zernike Moments based Fingerprint Matching

4.1 Zernike Moments

Moment descriptors have been studied for image analysis and recognition since 1960s [47][48][49][50][51]. Zernike introduced a set of complex polynomials which form a complete orthogonal set over the interior of a unit circle [52]. Teague investigated the image moments based on complex polynomials and the use of Zernike moments to overcome the shortcomings of information redundancy present in the geometric moments [48]. Zernike moments [47][53] are a class of orthogonal moments and possess a useful rotation invariance property. Rotating the image does not change the magnitude of the Zernike moments. These moments can be easily constructed to an arbitrary order. Although, higher order moments carry more fine details of an image, these are also more sensitive to noise. Therefore we have experimented with different orders of Zernike moments to determine the optimal order for our proposed fingerprint matching technique. The mathematical detail of Zernike moments is presented in section 4.4 of this chapter. Zernike moments and their corresponding number of features from order 0 to order 10 are shown in Table 4.1.

Order	Moments	No. of elements
0	Z_{00}	1
1	Z_{11}	1
2	Z_{20} , Z_{22}	2
3	Z_{31} , Z_{33}	2
4	Z_{40} , Z_{42} , Z_{44}	3
5	Z_{51} , Z_{53} , Z_{55}	3
6	Z_{60} , Z_{62} , Z_{64} , Z_{66}	4
7	Z_{71} , Z_{73} , Z_{75} , Z_{77}	4
8	Z_{80} , Z_{82} , Z_{84} , Z_{86} , Z_{88}	5
9	Z_{91} , Z_{93} , Z_{95} , Z_{97} , Z_{99}	5
10	$Z_{10,0}$, $Z_{10,2}$, $Z_{10,4}$, $Z_{10,6}$, $Z_{10,8}$, $Z_{10,10}$	6

Table 4.1: Zernike moments and their corresponding number of features from order 0 to order 10

4.2 Fingerprint Matching

Fingerprint matching techniques can be broadly classified into three main categories: minutiae based, image-based and hybrid. Minutiae-based fingerprint matching technique [54][55][56] first locates the minutiae points in a given fingerprint image and matches their relative placements in a stored template fingerprint. A good quality fingerprint contains between 40 and 60 minutiae, but different fingerprints have different number of minutiae. The performance of minutiae-based techniques rely heavily on the accurate detection of minutiae points and the use of complex matching techniques to compare two minutiae fields which undergo non-rigid transformations. Image-based fingerprint matching methods [46][57][58][59][60][61][62] use features such as texture information, ridge shape, ridge frequency and ridge orientation for matching. Hybrid fingerprint matching technique [63][64][65][71] is usually the combination of minutiae-based and image-based fingerprint matching techniques.

Hasen et. al. [79] proposed a fingerprint matching technique based on Zernike moments. Their technique relies on the accurate detection of core-point in a fingerprint image to form the Region of Interest (ROI). As core-point detection is not a trivial task and the core-point may be located along the sides of fingerprint image. Hence ROI may not be covering the ridge and valley structure of a fingerprint image properly. The core-point detected should be in the center of fingerprint image for proper formation of ROI. Also there is always some error in the accurate detection of core-point location. Our approach does not rely on the detection of core-point location and complete fingerprint image is utilized.

Most image-based matching techniques [46][57][63][72] rely heavily on the accurate detection of core-point location. Core point [70][73][74][75] is defined as the north most point of inner-most ridge line in a fingerprint. The region of interest (ROI) is defined around the core-point and then the image features are extracted from the region around the core-point. These techniques, however, suffers from the following shortcomings: (i) The ROI is based on a global singular point i.e., the core-point. Detection of the core point is non-trivial task as the core point may not even be present in small-sized images obtained using solid-state sensors. (ii) The ROI around the core-point does not cover the entire image. Furthermore, if the core was to be detected close to the boundary of the image, the ROI will include an extremely small portion of the image.(iii) The fingerprint alignment is based on a single core-point and is, therefore, not very robust with respect to errors in the location of the core-point.

4.3 Proposed Algorithm

Our proposed Zernike moment based fingerprint matching technique falls under the category of image-based matching techniques and its advantage over other techniques is that it does not rely on the core-point location as the ROI is the complete fingerprint image in the frequency domain. Our proposed fingerprint matching technique has the following steps:

i. Fingerprint image is enhanced using Gabor filter based technique.
ii. Fingerprint image is converted into a binary image.
iii. Binary fingerprint image is converted into frequency domain by taking its Discrete Fourier Transform (DFT).
iv. Absolute of DFT is taken.

v. Zernike moments are calculated up to order 50.

vi. Magnitude of Zernike moments is calculated.

vii. Normalized Euclidean distance is calculated between the template and query image.

4.3.1 Fingerprint Enhancement

Ling Hong [66] proposed the Gabor filter based fingerprint image enhancement technique. Gabor filters were utilized to extract the ridges from the fingerprint image. In this technique the fingerprint image was divided into non-overlapping blocks. Then the ridge frequency and ridge orientation of each block was calculated. Then these blocks were filtered with Gabor filters which were tuned to the frequency and orientation of each block. Ridges were extracted from each block. Original fingerprint image and enhanced image is shown in Figure 4.1 and Figure 4.2.

Figure 4.1: Original fingerprint image

Figure 4.2: Enhanced image using Gabor enhancement technique

Figure 4.3: Enhanced image converted into binary image

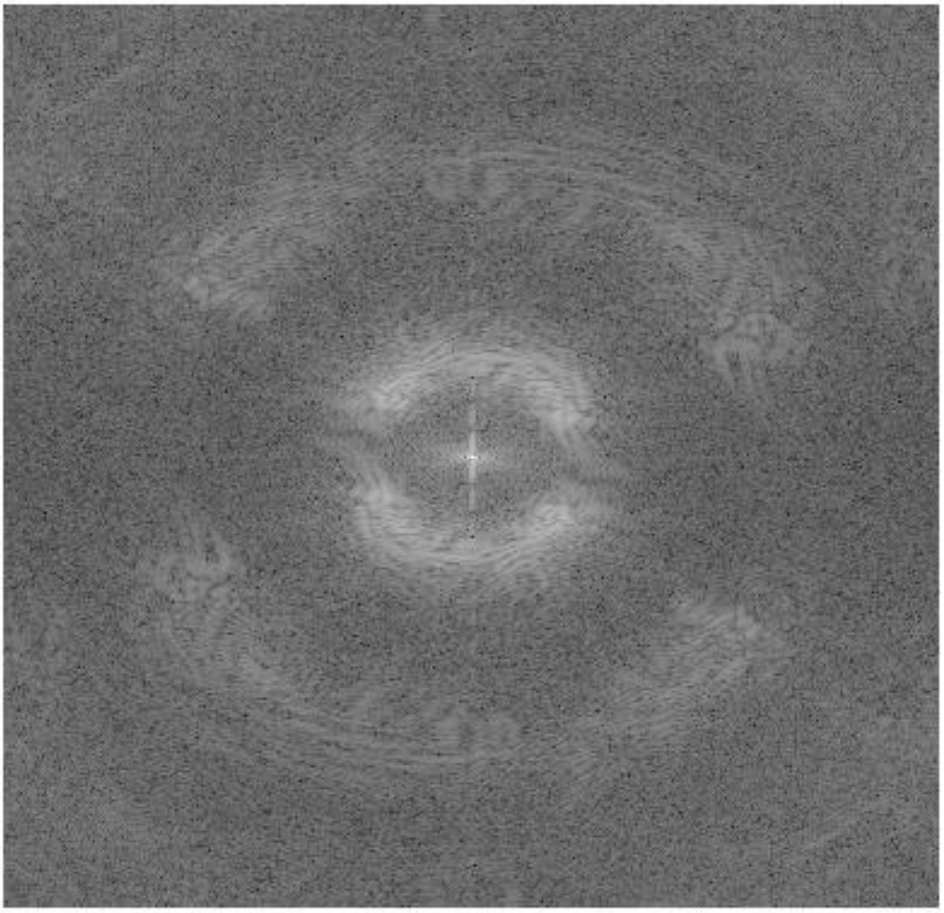

Figure 4.4: Spectrum of Discrete Fourier Transform of binary image

4.3.2 Spectrum of DFT

The enhanced Gabor image was converted into a binary image as shown in Figure 4.3. Then this binary image was converted into the frequency domain by taking its Discrete Fourier Transform (DFT) [67]. Then the spectrum of DFT which is shown in Figure 4.4 was obtained by taking absolute of DFT. The spectrum of DFT was invariant to translation due to shift property of DFT [53][68].

4.4 Feature Extraction based on Zernike moments

For fingerprint matching, it is desirable to obtain the fingerprint features which are scale, translation and rotation invariant. Scale invariance is not a major problem as it is dependent on the dots per inch (dpi) specification of a particular fingerprint scanner. Therefore the fingerprint images acquired through the particular scanner could be scaled as per the dpi specification of that scanner. Translation invariance is achieved by converting the fingerprint image into frequency domain. The translation property of Fourier transform states that: when there is a translation in spatial domain, only the phase changes in the frequency domain, the magnitude does not change [53][68]. Therefore, the magnitude of the DFT which is the absolute values of the DFT coefficients was calculated which is invariant to translation. To achieve the rotation invariance, Zernike moments were calculated from the DFT of the fingerprint image as these moments are rotation invariant. Zernike moments [47][48] are a set of complex polynomials, which form a complex orthogonal set over the interior of a unit circle. i.e. $x^2 + y^2 = 1$. Let the set of these Zernike polynomials denoted by $Z_{nm}(x,y)$ defined in equation 4.1.

$$Z_{nm}(x,y) = Z_{nm}(\rho,\theta) = R_{nm}(\rho)e^{jm\theta} \qquad (4.1)$$

n = positive integer or zero

m= positive and negative integers subject to constraints n- |m| is even, and |m| is less than or equal to n

p = Length of vector from origin to (x,y) pixel

θ = Angle between vector ***p*** and **x-**axis in counterclockwise direction

Rnm (ρ) is a radial polynomial which is defined as:

$$R_{nm}(\rho) = \sum_{s=0}^{\frac{n-|m|}{2}} (-1)^s \frac{(n-s)!}{s!\left(\frac{n+|m|}{2}-s\right)!\left(\frac{n-|m|}{2}-s\right)!}\rho^{n-2s} \qquad (4.2)$$

The magnitude of Zernike moments are invariant to image rotation and can be used as rotation invariant feature. The Zernike moment Z'_{nm} of a rotated image is simply related to that Z_{nm} of the un-rotated image by phase change proportional to the degree of rotation as given in equation 4.3.

$$Z'_{nm} = Z_{nm}\, e^{jm\theta} \qquad (4.3)$$

Zernike moments were extracted from the spectrum of DFT of a fingerprint image. Zernike moments have been calculated up to order 50. It has been shown through experimental results

that order of Zernike moments varies for different type of fingerprint databases to achieve the minimum error rate.

4.5 Normalized Euclidean distance based matching

Normalized Euclidean distance based classification is used for matching of fingerprints. The normalization is done on each moment feature to account for variance in that feature dimension. Mean, standard deviation and normalized Euclidean distance was calculated using the equations given below:

$$\mu_i^k = \frac{1}{n}\sum_{j=1}^{n} x_{i,j}^k \tag{4.4}$$

$$\sigma_i^k = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(x_{i,j}^k - \mu_i^k)^2} \tag{4.5}$$

$$d(x, C_k) = \sum_{i=1}^{m}\left(\frac{x_i - \mu_i^k}{\sigma_i^k}\right)^2 \tag{4.6}$$

μ_i^k = Mean of the ith feature in class k

σ_i^k = Standard deviation of the ith feature in class k

$x_{i,j}^k$ = Value of the ith feature of sample j in class k

$d(x, C_k)$ = Normalized Euclidean distance between sample x and class k

n = Number of samples in class k

m = Feature dimension

For each fingerprint in the database, six samples were used in the training and remaining two samples were used for testing. In the training phase, a single normalized template has been generated from the six samples of each fingerprint. Fingerprint matching is based on finding the normalized Euclidean distance between the normalized template and Zernike features of query fingerprint image.

4.5 Summary

In this chapter, a novel fingerprint matching algorithm based on Zernike moments is proposed. For fingerprint matching, it is desirable to obtain a fingerprint representation invariant to translation and rotation. Translation invariance is achieved by transforming the fingerprint image into frequency domain and taking the absolute yielding the spectrum of an image invariant to translation. For rotation invariance, Zernike moments are calculated which are invariant to rotation. The fingerprint image is first enhanced and then converted into frequency domain by taking its Discrete Fourier Transform. Then the magnitude of the Zernike moments is calculated. The fingerprint matching is based on the normalized Euclidean distance between the two corresponding Zernike moments of stored template and query fingerprint image. Experimental results are discussed in chapter 5 which shows that the proposed method has better performance in terms of matching accuracy as compared to the traditional Gabor filter based fingerprint matching methods.

Chapter 5

Experiment Results and Analysis

5.1 Experiment Results

The Zernike moment based fingerprint matching technique has been tested on Fingerprint Verification Competition (FVC) 2002 [69] databases which are database 1 (Db1), database 2 (Db2), database 3 (Db3) and database 4 (Db4). Each database contains 800 fingerprints (100 fingerprint classes, 8 impressions per class).

In a fingerprint matching system, there are four possible outcomes:

a. Genuine Acceptance
b. Imposter Rejection
c. Genuine Rejection (False Rejection)
d. Imposter Acceptance (False Acceptance)

The outcomes defined in (a) and (b) are correct while the outcomes defined in (c) and (d) are errors. The performance of a fingerprint matching system is given in terms of false accept rate (FAR) and false reject rate (FRR). The equal error rate is used as a performance measure. The Equal Error Rate (EER) indicates the point where FAR and FRR are equal. If the normalized Euclidean distance between the template feature and query feature is less than a threshold, then

the decision that “the two fingerprints are matched” is made, otherwise a decision that “the two fingerprints are not matched” is made.

The experiments were performed on Intel Core 2 Duo 2.4 GHz processor running Microsoft Windows XP. The algorithms were implemented on a MATLAB. In these experiments, each database was divided into a training set and testing set. In training phase, six out of eight fingerprints from each class were chosen and a single normalized feature vector template was generated for each class. In total 100 templates were generated in training phase for each database. In testing phase 200 fingerprints were used. These were those fingerprints which were not used in training. For each database in FVC 2002, a total of 200 genuine matches and 4950 imposter matchers were performed.

To compute FAR and FRR, the genuine matches and imposter matches were performed on the four testing sets of FVC2002 database. In genuine matching each fingerprint of each class is compared with other fingerprints of same class. For imposter matching, the fingerprint of each class was compared with the fingerprints belonged to other classes. As there were 200 test fingerprints, the total number of genuine matches were 100 x 2 = 200 and total number of imposter matches were 4950 (100 x 99 / 2) for each database.

For comparison we have implemented the Gabor filter bank based matching technique [46][73][74]. This technique depends on the accurate detection of core point location which should be in the center of fingerprint. All those images in which core point was detected at the sides of fingerprint images were not used in matching. Total number of genuine matches varies

from 2600 to 2700 for each database while the total number of imposter matches was 4950 for each database. Curves of FAR & FRR versus threshold values and Receiver Operating Curves of Gabor filter based fingerprint matcher tested on FVC 2002 db1, db2, db3 and db4 databases are shown in Figure 5.1 to Figure 5.8.

The Zernike moments up to order 50 have been calculated. There were total of 676 Zernike feature extracted from each fingerprint. EER was computed for different number of Zernike features. Experiments were performed by varying the number of Zernike features from 100 to 650 to get the best EER for each database. It has been observed that the best EER was obtained for each database by using different number of Zernike features. Curves of FAR & FRR versus threshold values and Receiver Operating Curves (ROC) of Zernike moments based fingerprint matcher tested on FVC 2002 db1, db2, db3 and db4 databases are shown in Figure 5.9 to Figure 5.142.

These ROCs of Zernike moments have been used to analyze the effects of varying the number of moment features on fingerprint matching performance. A series of experiments have been conducted by varying the number of Zernike features and their effect was studied on matching performance. To best of our knowledge, these experiments have not been done before in fingerprint matching as only fixed number of Zernike features were used by other researcher in matching of fingerprints. They did not analyze the effect of utilizing different number of moment features. It has been shown through these experiments that for different type of fingerprint databases, number of Zernike features were not same to obtain the best matching results. Best matching performance has been obtained by using the different number of Zernike features for

each database. As each database was obtained using different type of fingerprint scanners, so this is a major factor in deciding the optimal number of Zernike features for each database.

For FVC 2002 Db1, the minimum EER of 14.99 is achieved when 530 Zernike features were used in matching. For FVC 2002 Db2, the minimum EER of 18.49 is achieved when number of Zernike features used in matching was 398. The minimum EER obtained for Db3 is 9.48 using 370 Zernike features and for Db4, the minimum EER is 11.98 using 475 Zernike features. The complete experimental results of Db1, Db2, Db3 and Db4 are given in Table 5.1, 5.2, 5.3 and 5.4.

FVC 2002 Db1	
Zernike Features	**EER (%)**
200	22.87
250	22.61
300	22.12
350	20.07
370	19.04
400	16.79
420	17.00
450	16.08
480	16.50
500	16.41
520	15.48
530	14.99
540	14.99
550	16.01
560	16.00
580	16.50
600	16.99

Table 5.1: EER calculated on FVC 2002 Db1 using different number of Zernike moments. The minimum EER obtained for each database is highlighted.

FVC 2002 Db2	
Zernike Features	**EER (%)**
100	23.00
150	20.87
200	19.40
250	19.09
300	20.00
350	19.50
370	18.99
398	18.49
400	18.50
420	18.99
450	18.49
480	18.50
500	18.88
520	18.60
550	18.50
580	18.50
600	18.49

Table 5.2: EER calculated on FVC 2002 Db2 using different number of Zernike moments. The minimum EER obtained for each database is highlighted.

FVC 2002 Db3	
Zernike Features	**EER (%)**
245	10.51
250	10.99
260	10.93
270	10.96
300	10.06
350	9.79
360	9.52
370	9.48
380	9.50
390	9.56
400	9.94
425	10.00
450	10.00

Table 5.3: EER calculated on FVC 2002 Db3 using different number of Zernike moments. The minimum EER obtained for each database is highlighted.

FVC 2002 Db4	
Zernike Features	**EER (%)**
100	15.07
150	15.47
200	14.08
250	12.99
300	12.99
350	12.50
398	12.89
400	12.62
420	12.90
450	12.09
470	12.42
475	11.98
480	12.00
490	12.00
500	12.00
520	12.08
550	12.00
570	12.47

Table 5.4: EER calculated on FVC 2002 Db4 using different number of Zernike moments. The minimum EER obtained for each database is highlighted.

5.2 Analysis

The results of our proposed method are better than the Gabor filter bank based matching technique on all the four databases of FVC 2002 as shown in Table 5.5. Our proposed method has out-performed the Gabor filter bank based matching technique on all four databases. The Fingercode Matcher technique [59] performed better on Db1 and Db2 database but this technique was tested on 600 images instead of 800 images as reported by the authors. In Fingercode matcher method, those fingerprint images were excluded from the database in which core point lies on the side of the fingerprint image or the fingerprint image was highly displaced. The performance of Fingercode matcher technique was better due to the exclusion of 200 fingerprint images from each FVC 2002 database. The performance of our technique was better on Db3 and Db4 although it was tested on complete 800 images of Db3 and DB4. The time taken by the feature extraction stage in our technique varies between 8 ~ 9 seconds. This was an optimized time using Zernike masks for feature extraction for a single fingerprint image. This time was dependent on the size of the fingerprint image as large size images took more time as compared to smaller images. This feature extraction time can be cut down using reduced size images. The matching time is approximately 1~2 milliseconds for each fingerprint query image.

		Db1	**Db2**	**Db3**	**Db4**
1	**Gabor Filter based Matcher**	27.55	36.8	37.64	25.18
2	**Fingercode Matcher**	12.5	11.7	29	18
3	**Zernike Moments based Fingerprint Matcher**	14.99	18.49	9.48	11.98

Table 5.5: EER comparison of fingerprint matching techniques on all the four databases of FVC 2002

5.3 Summary

We have proposed a novel fingerprint matching technique using Zernike moments. Different number of Zernike features has been tested on FVC 2002 Db1,Db2, Db3 and Db4 databases. The number of Zernike features used in fingerprint matching varies for each database to obtain the minimum EER. In our proposed algorithm, translation is handled by taking the magnitude of the DFT of the fingerprint image which is translation invariant. Zernike moments exhibit rotation invariance property. Therefore Zernike features extracted were rotationally invariant. The performance of our proposed algorithm is very good as it performed better as compare to other matching schemes.

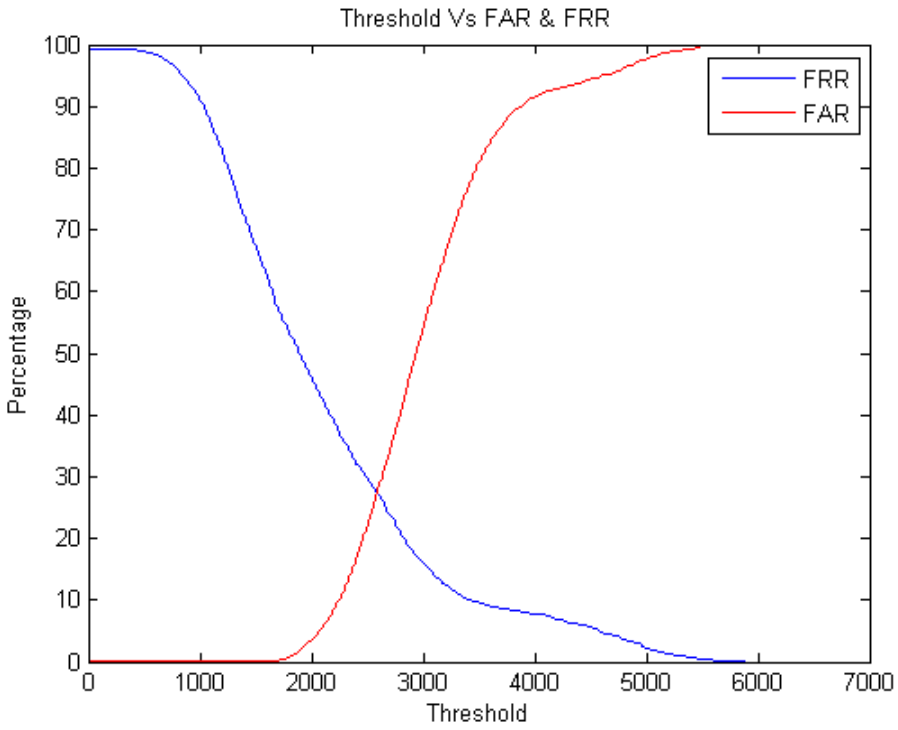

Figure 5.1: EER of Gabor Filter based Matcher tested on DB1

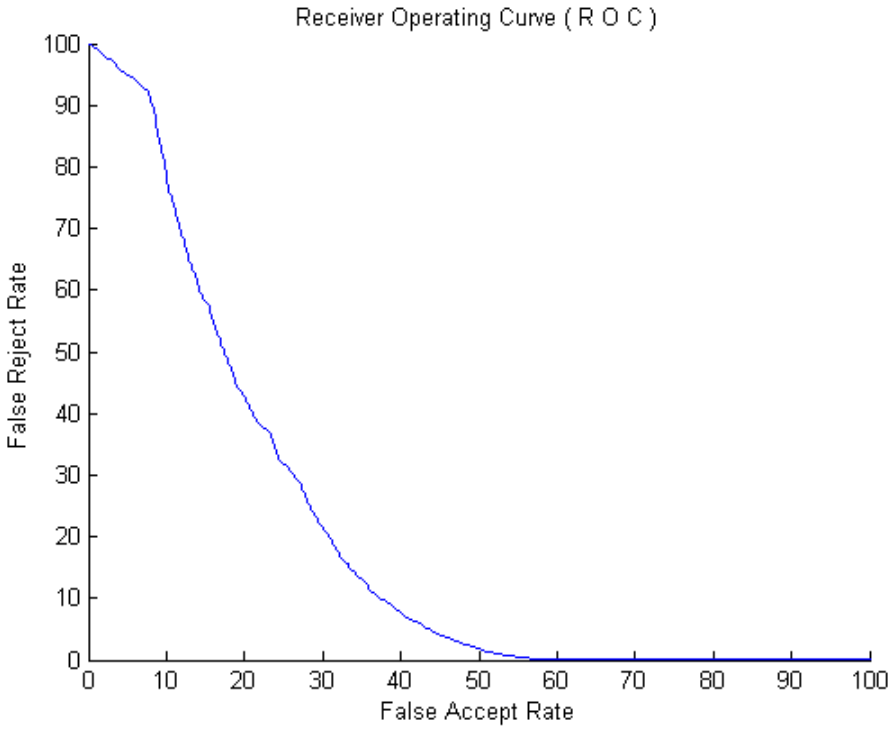

Figure 5.2: ROC Curve of Gabor Filter based Matcher tested on DB1

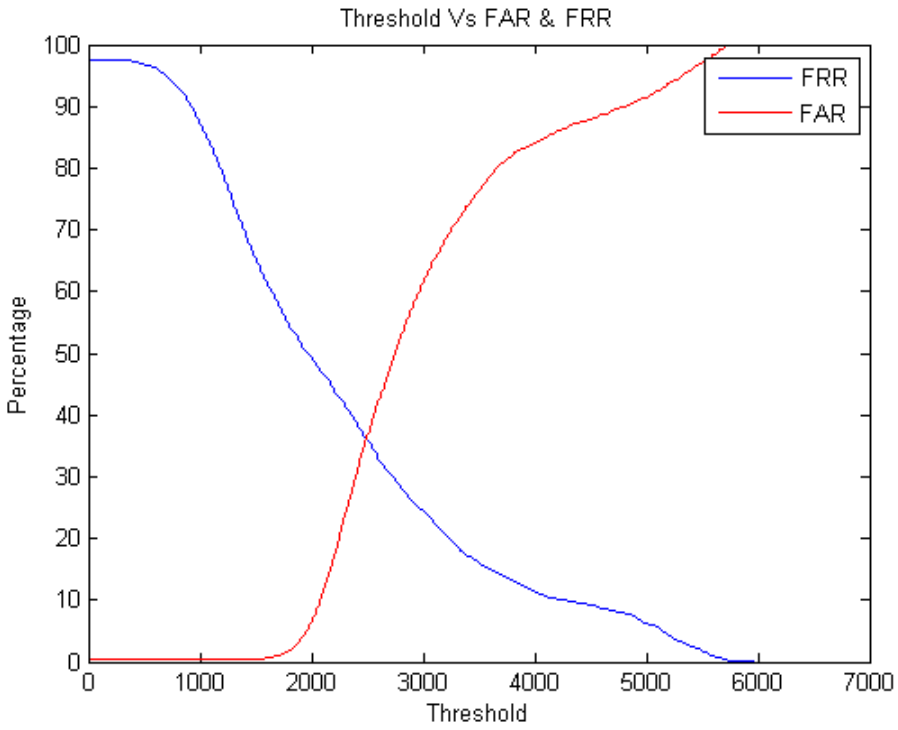

Figure 5.3: EER of Gabor Filter based Matcher tested on DB2

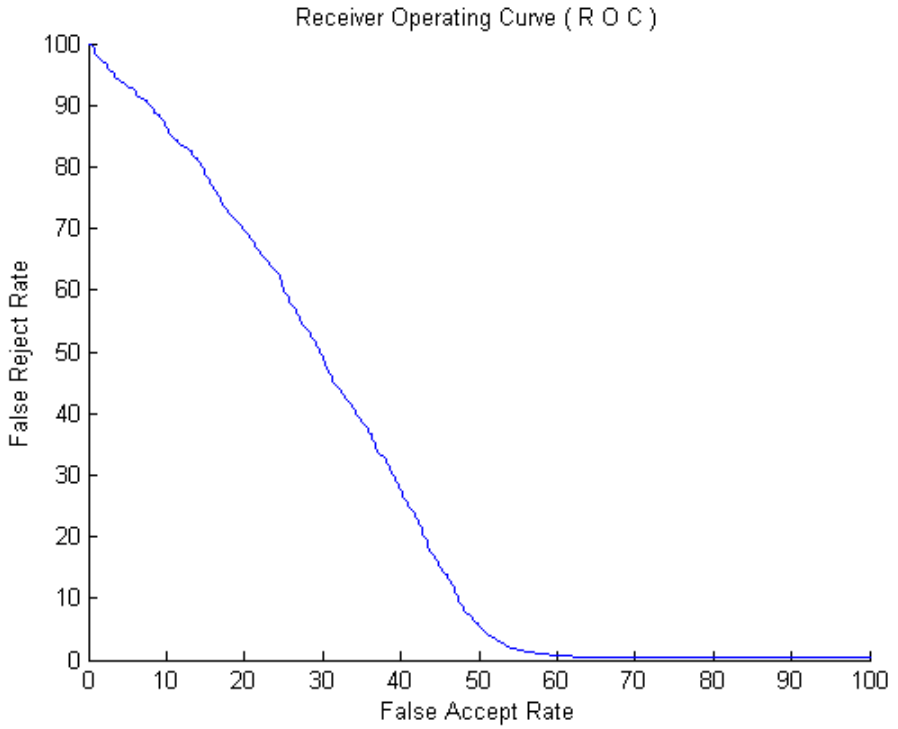

Figure 5.4: ROC Curve of Gabor Filter based Matcher tested on DB2

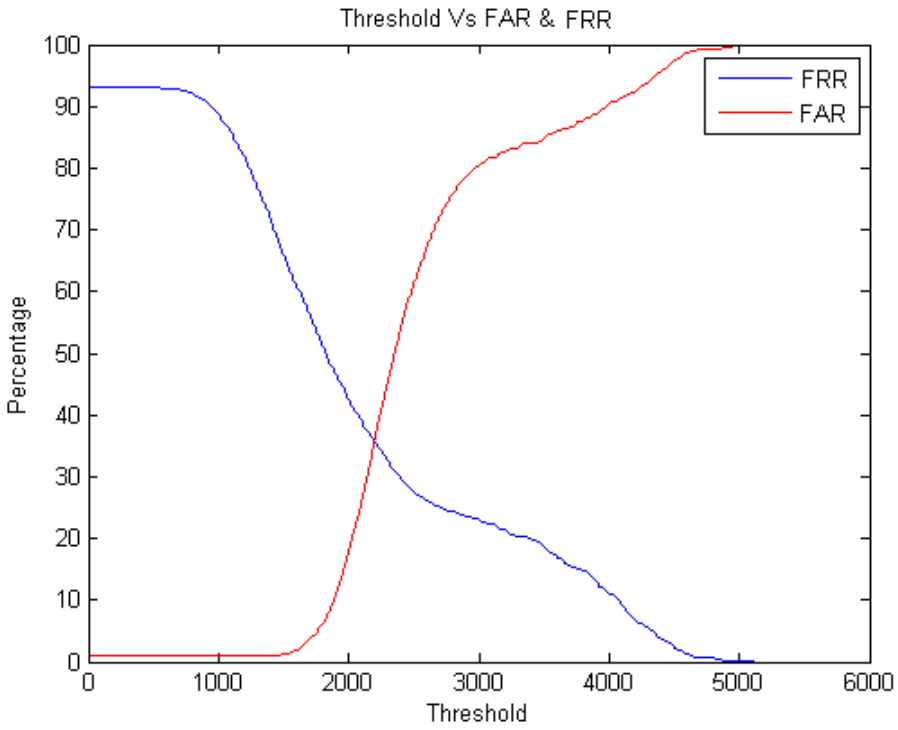

Figure 5.5: EER of Gabor Filter based Matcher tested on DB3

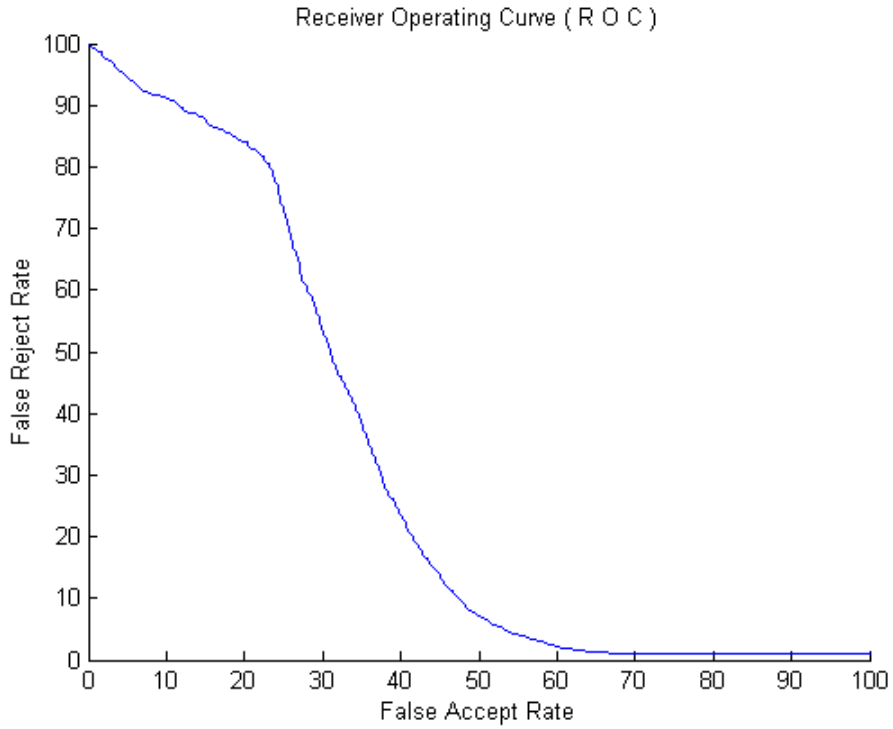

Figure 5.6: ROC Curve of Gabor Filter based Matcher tested on DB3

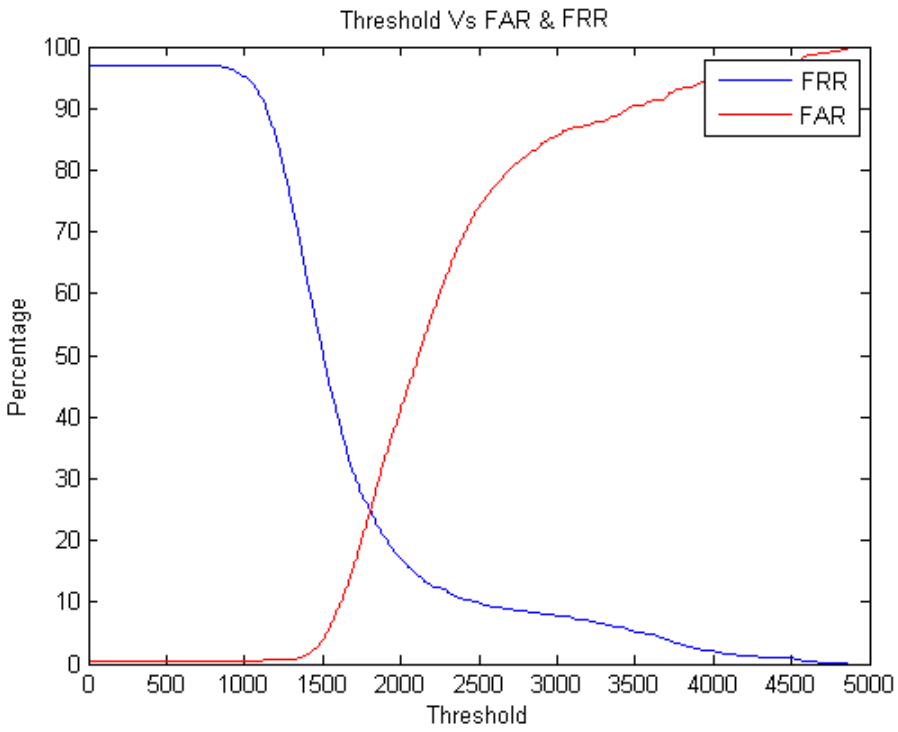

Figure 5.7: EER of Gabor Filter based Matcher tested on DB4

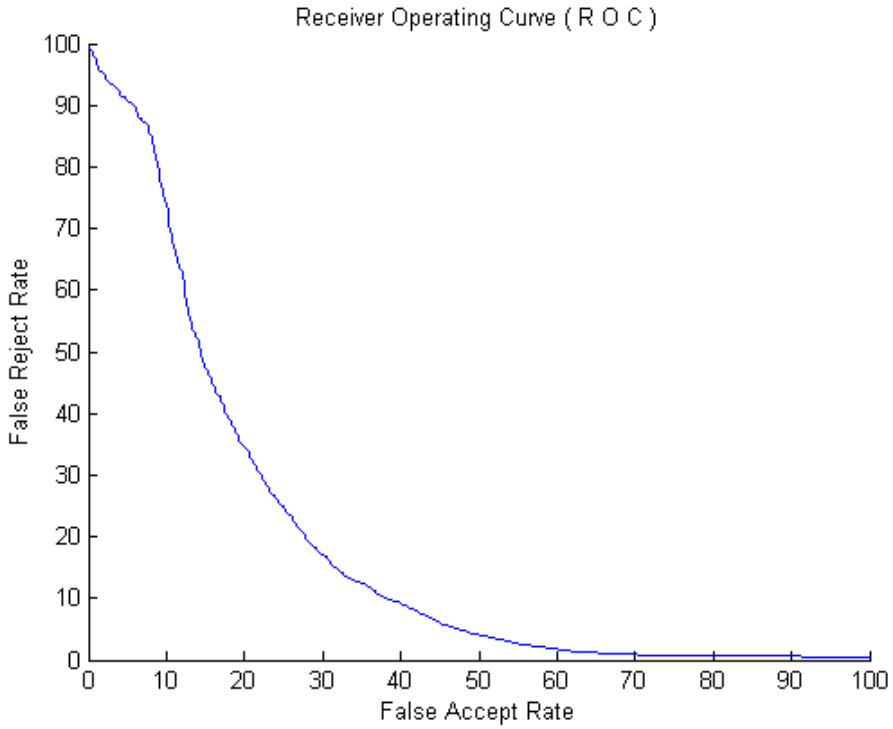

Figure 5.8: ROC Curve of Gabor Filter based Matcher tested on DB4

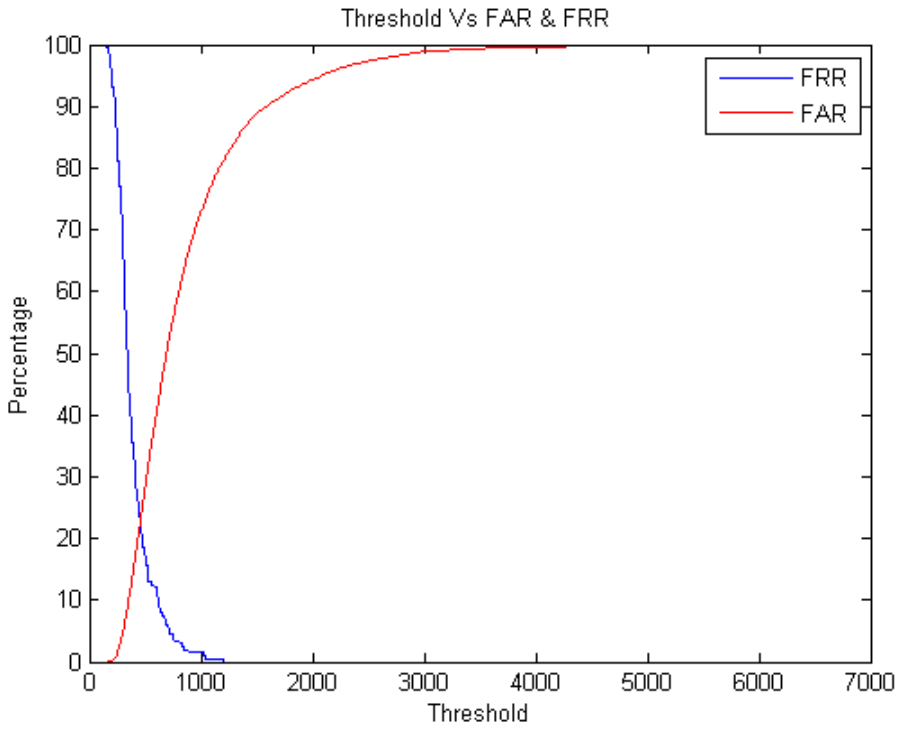

Figure 5.9: EER of 200 Zernike features tested on DB1

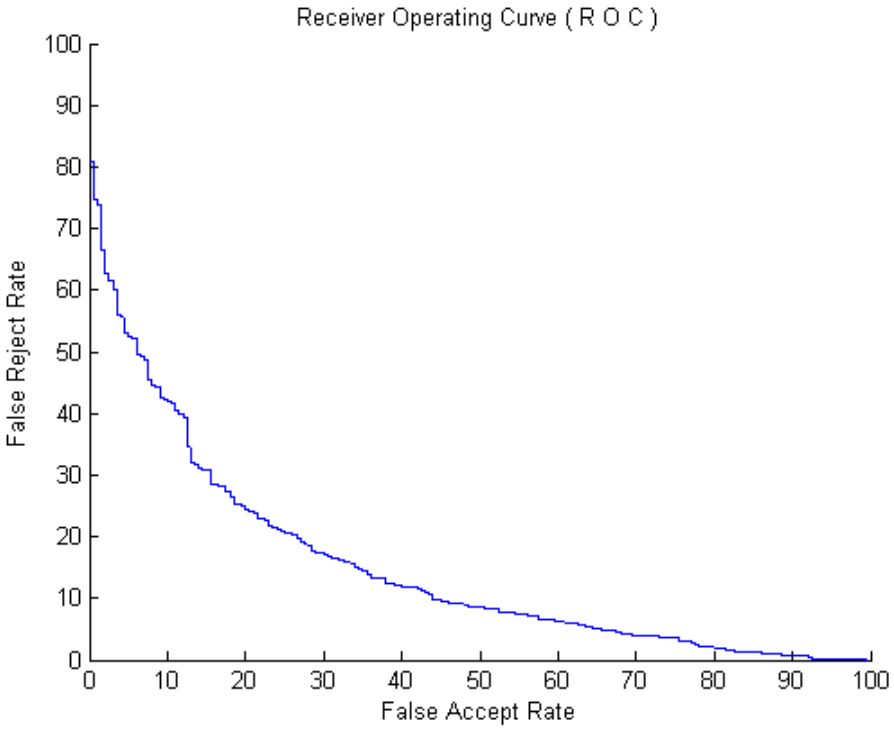

Figure 5.10: ROC Curve of 200 Zernike features tested on DB1

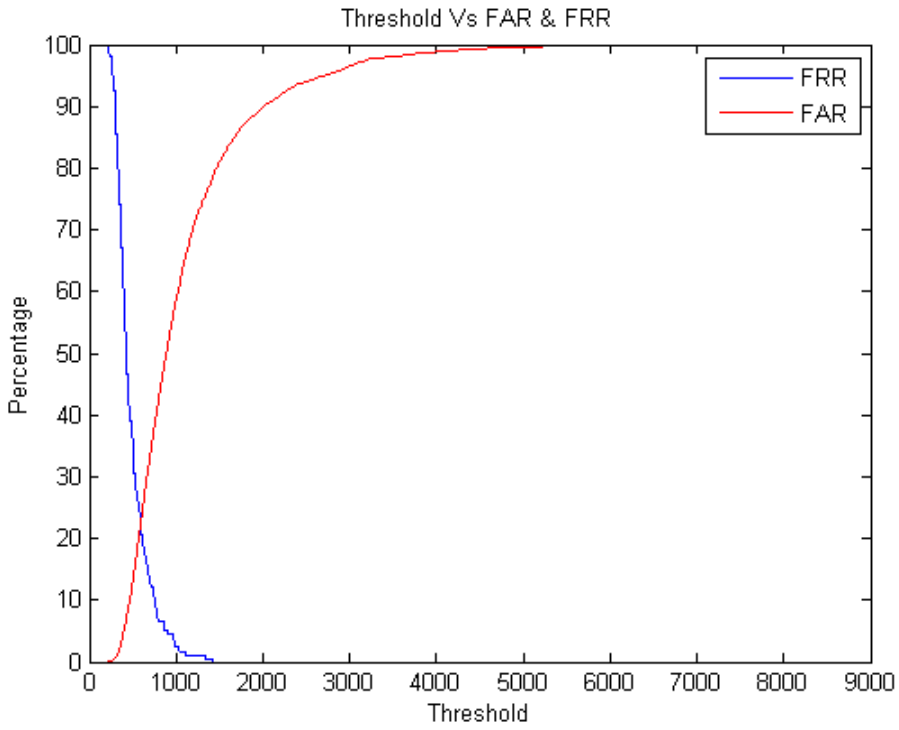

Figure 5.11: EER of 250 Zernike features tested on DB1

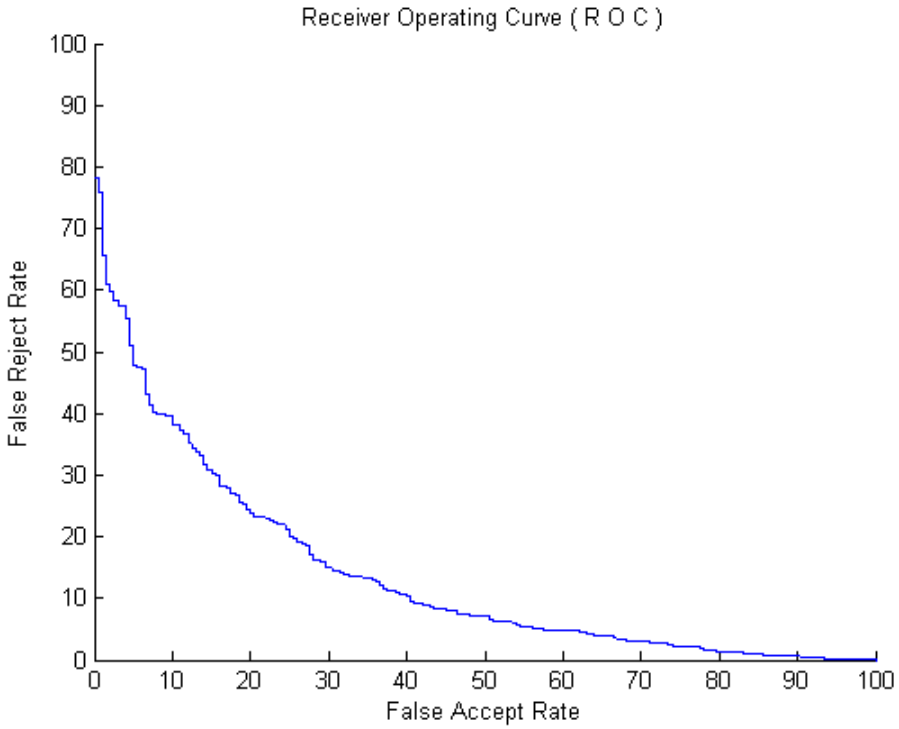

Figure 5.12: ROC Curve of 250 Zernike features tested on DB1

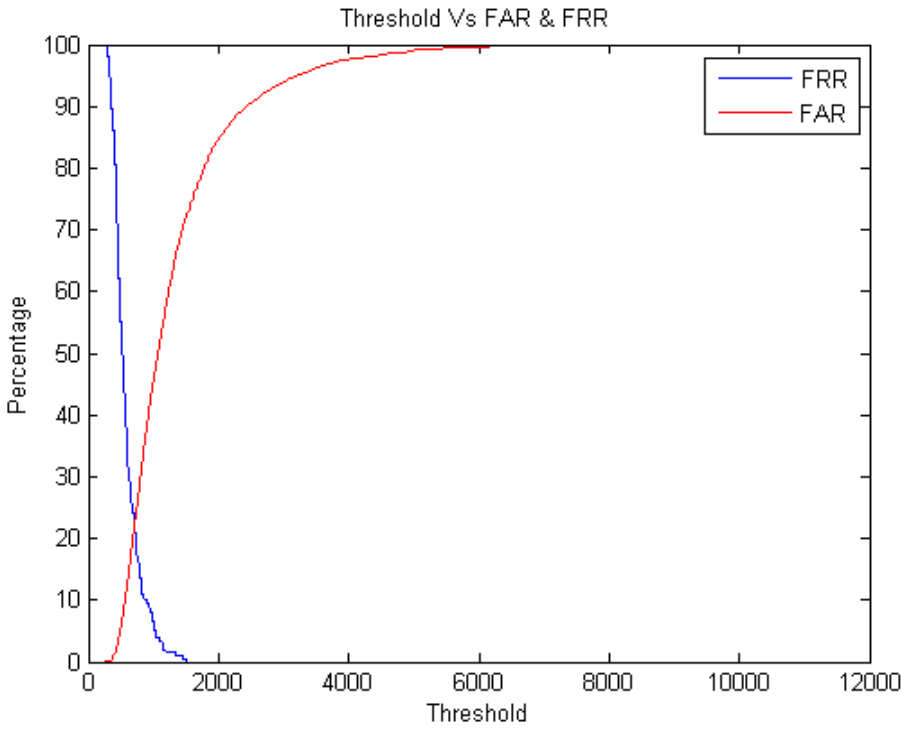

Figure 5.13: EER of 300 Zernike features tested on DB1

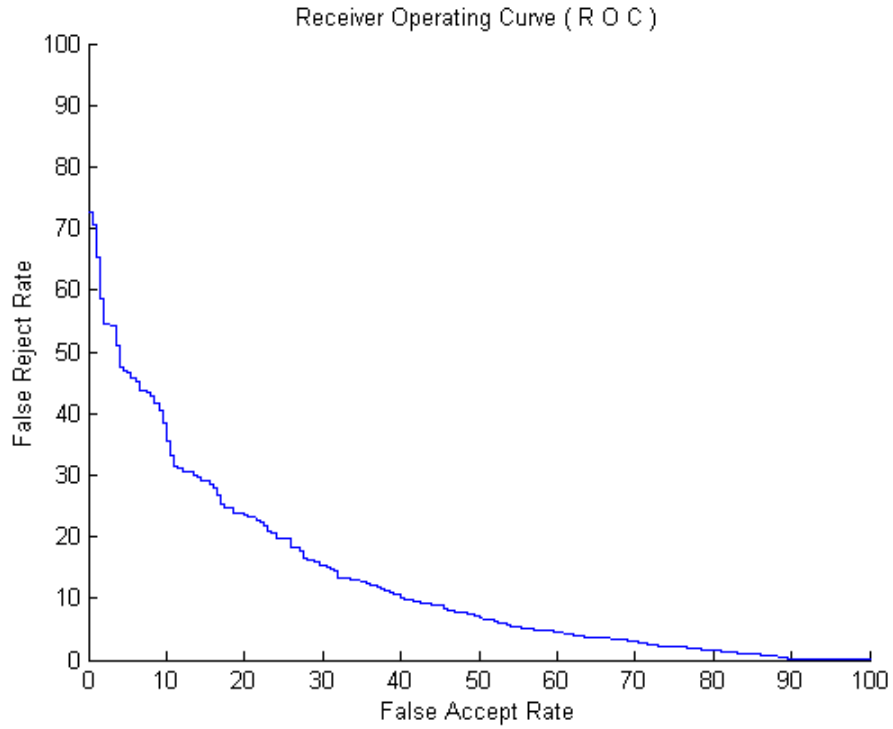

Figure 5.14: ROC Curve of 300 Zernike features tested on DB1

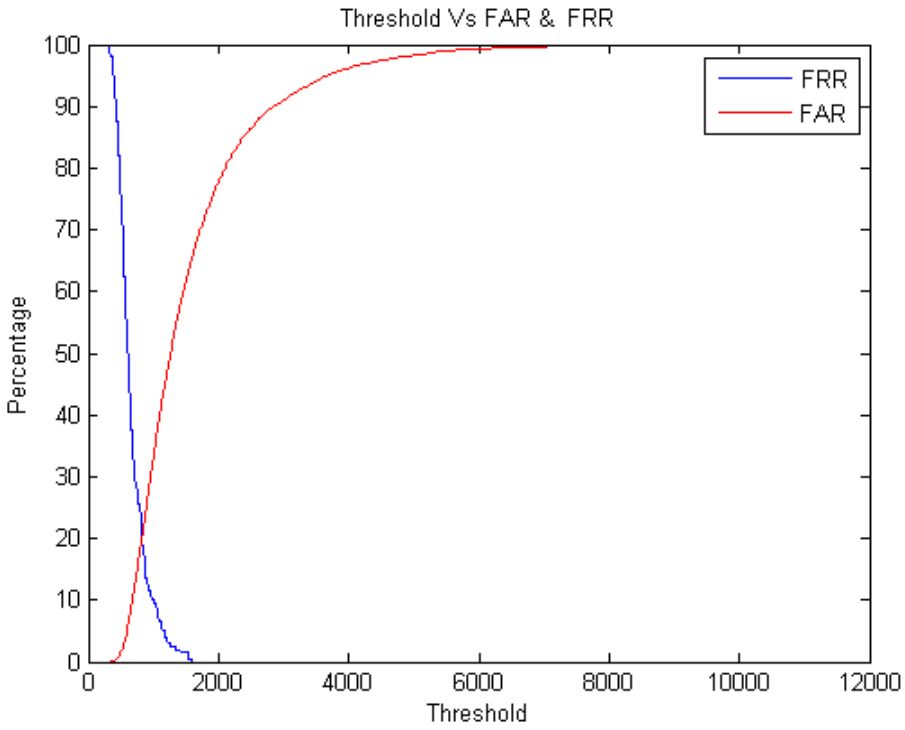

Figure 5.15: EER of 350 Zernike features tested on DB1

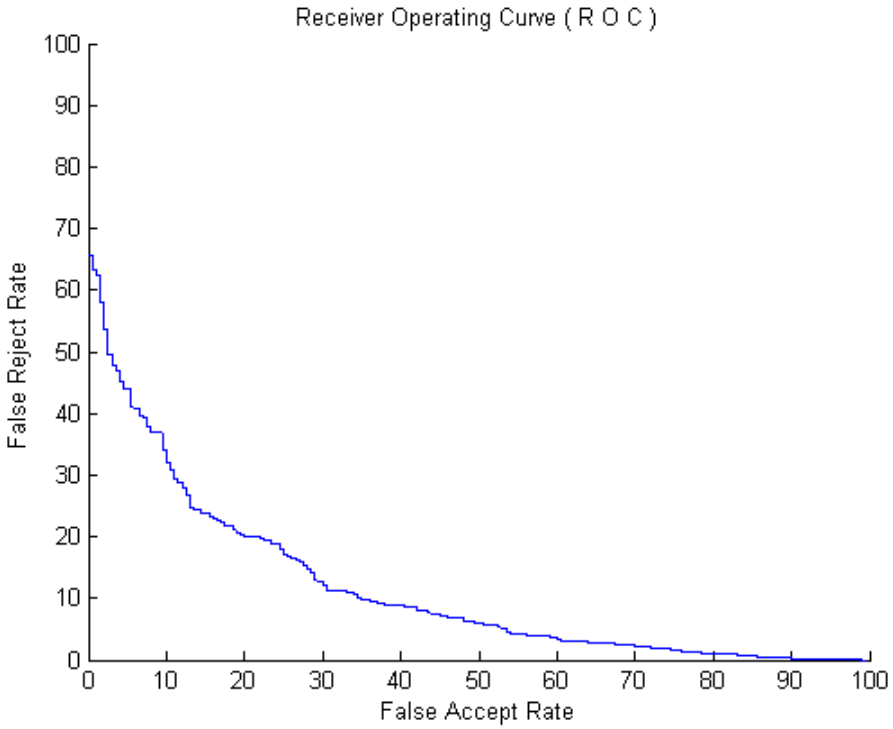

Figure 5.16: ROC Curve of 350 Zernike features tested on DB1

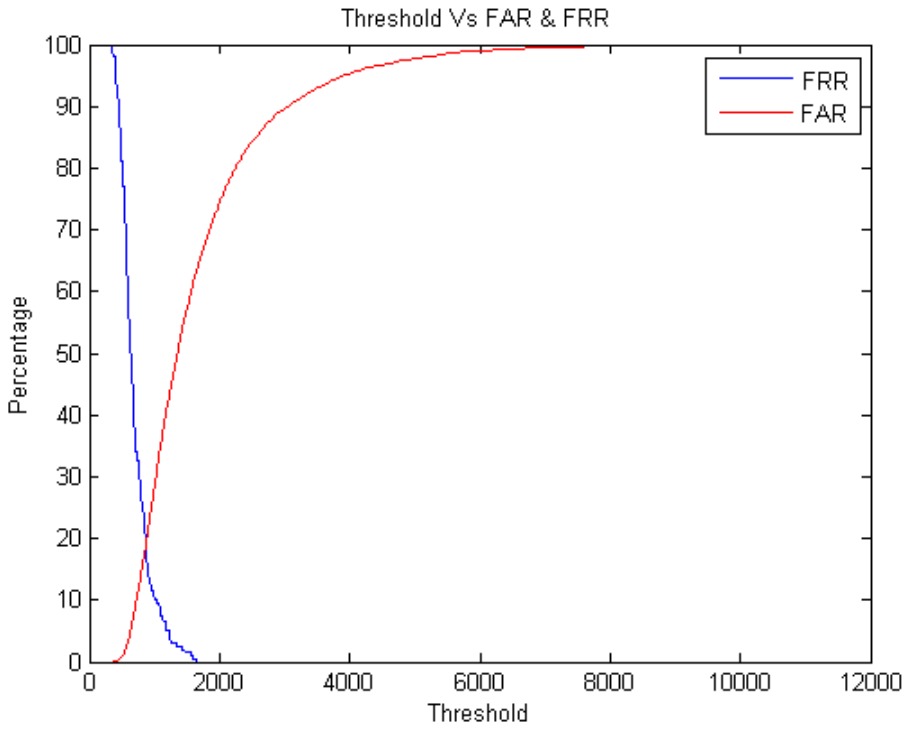

Figure 5.17: EER of 370 Zernike features tested on DB1

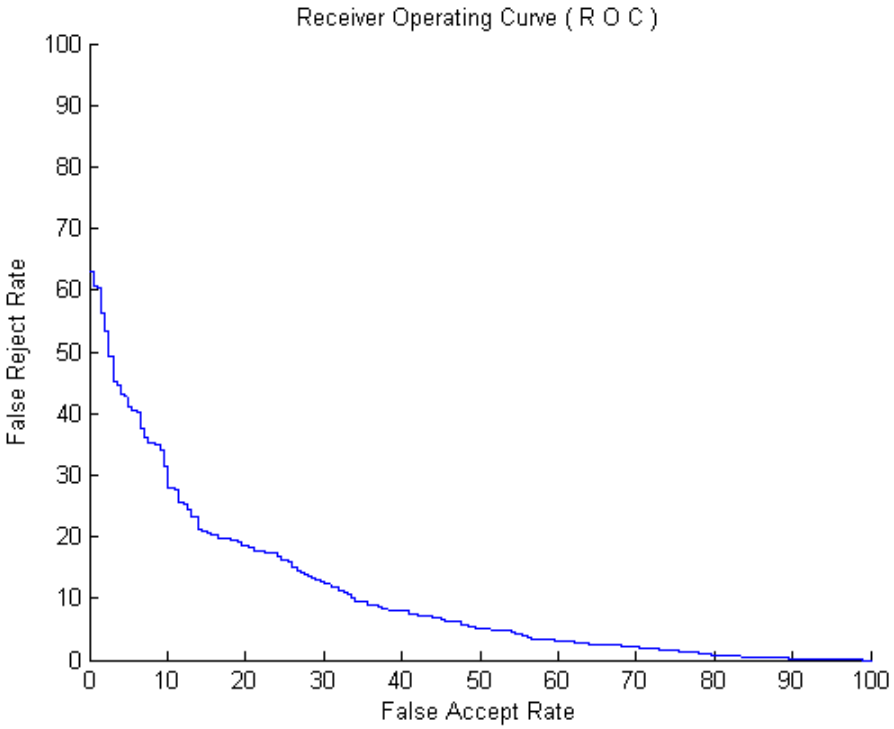

Figure 5.18: ROC Curve of 370 Zernike features tested on DB1

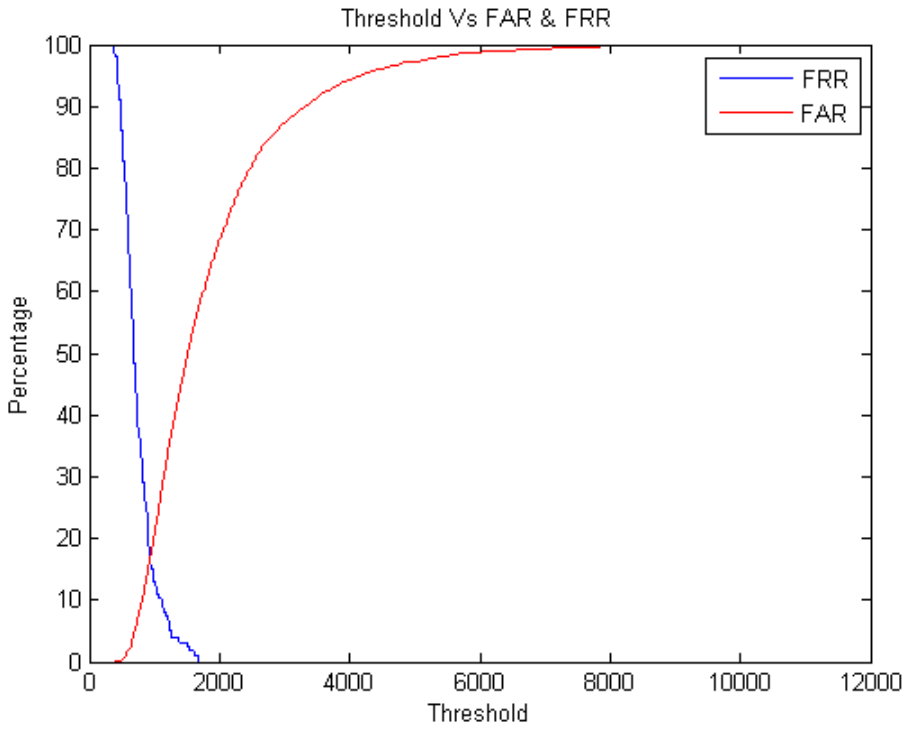

Figure 5.19: EER of 400 Zernike features tested on DB1

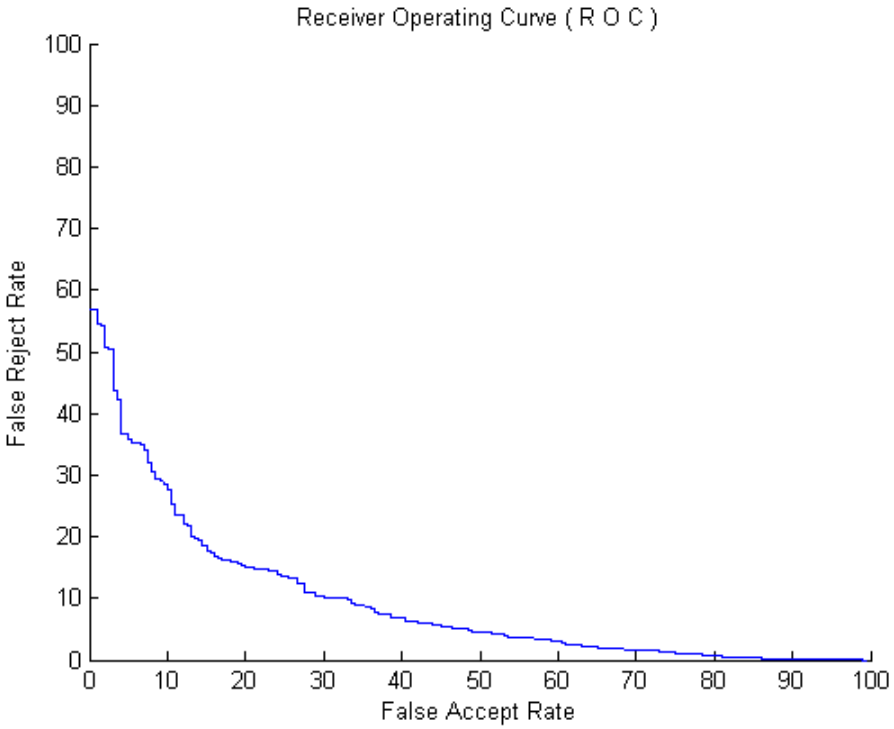

Figure 5.20: ROC Curve of 400 Zernike features tested on DB1

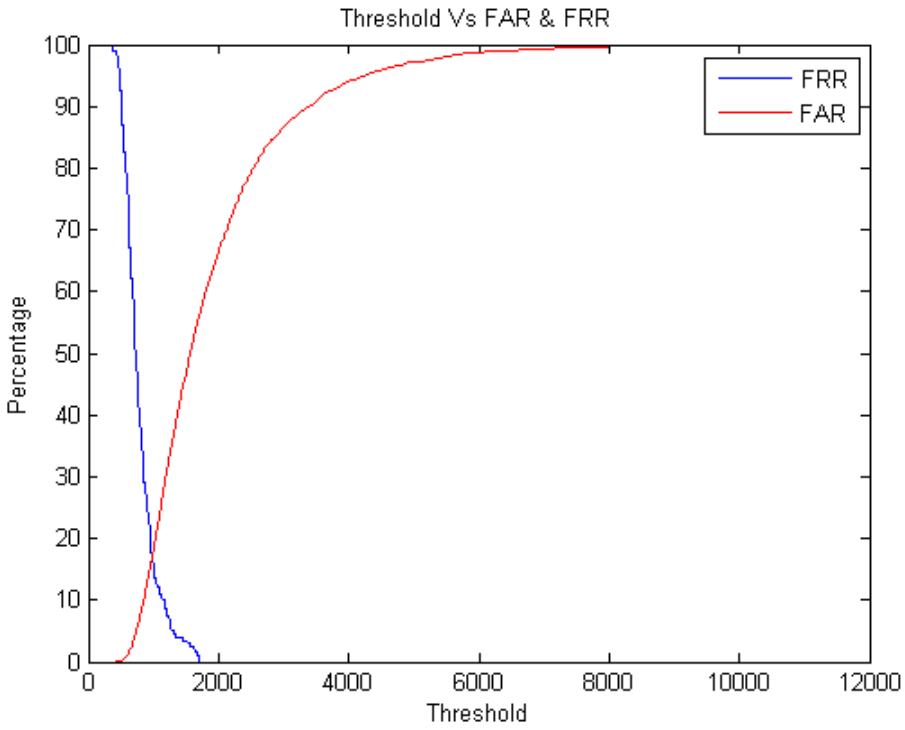

Figure 5.21: EER of 420 Zernike features tested on DB1

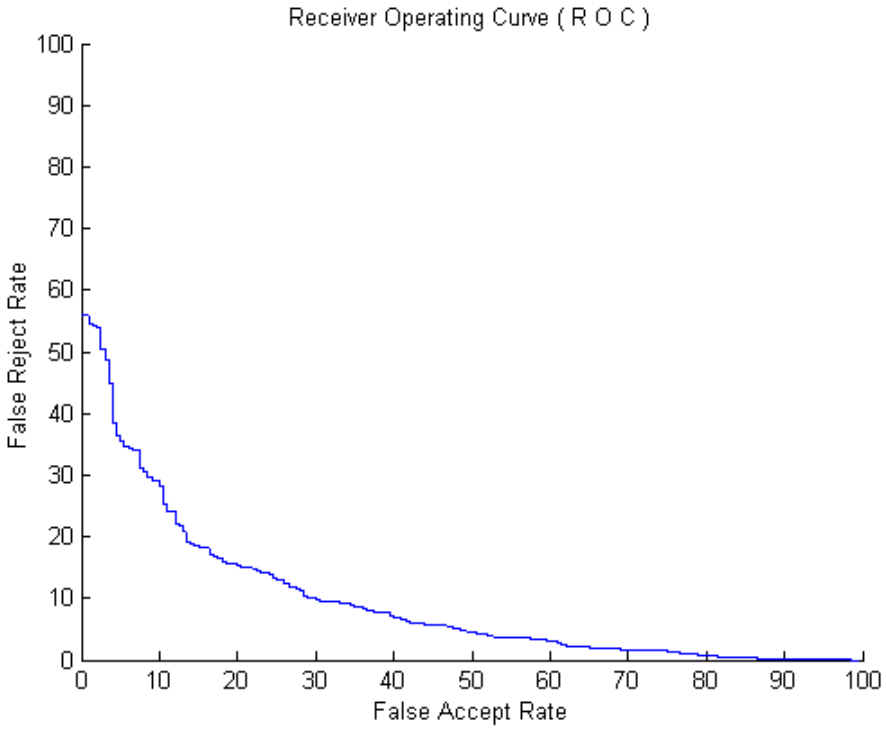

Figure 5.22: ROC Curve of 420 Zernike features tested on DB1

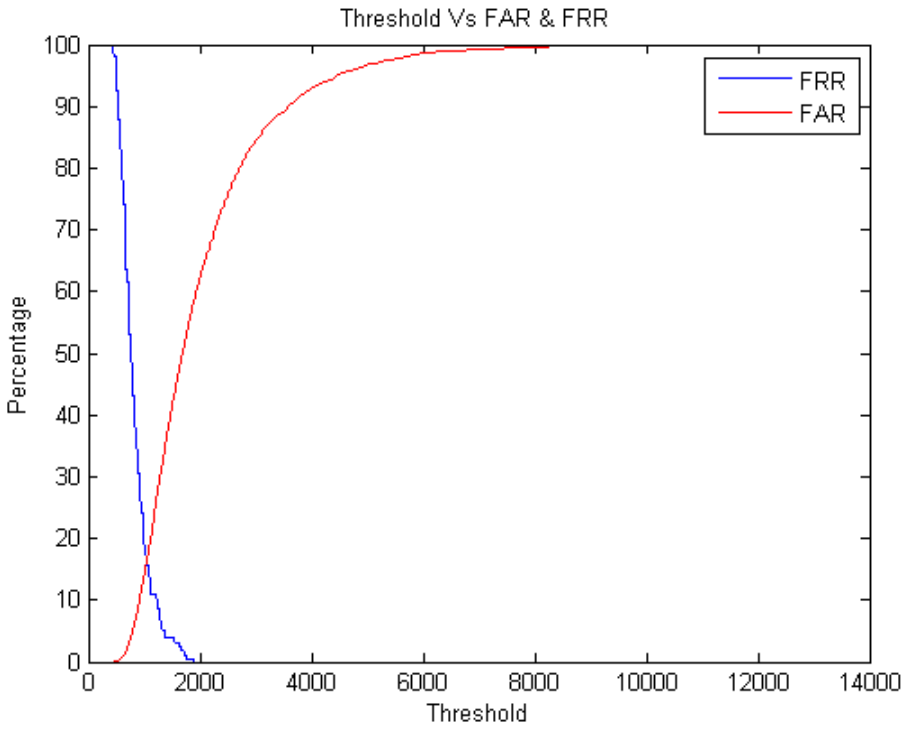

Figure 5.23: EER of 450 Zernike features tested on DB1

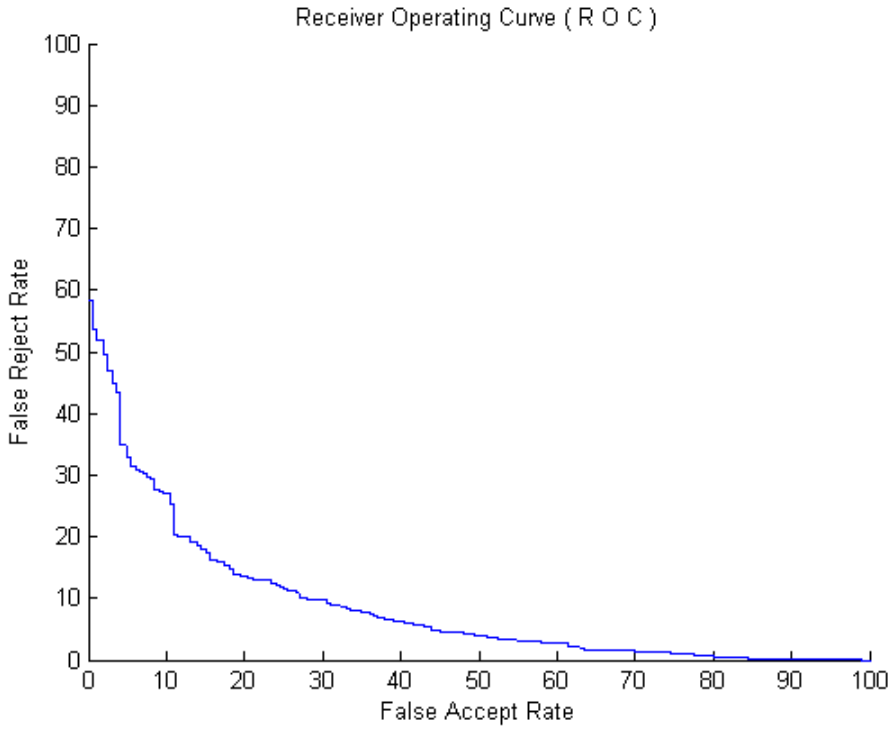

Figure 5.24: ROC Curve of 450 Zernike features tested on DB1

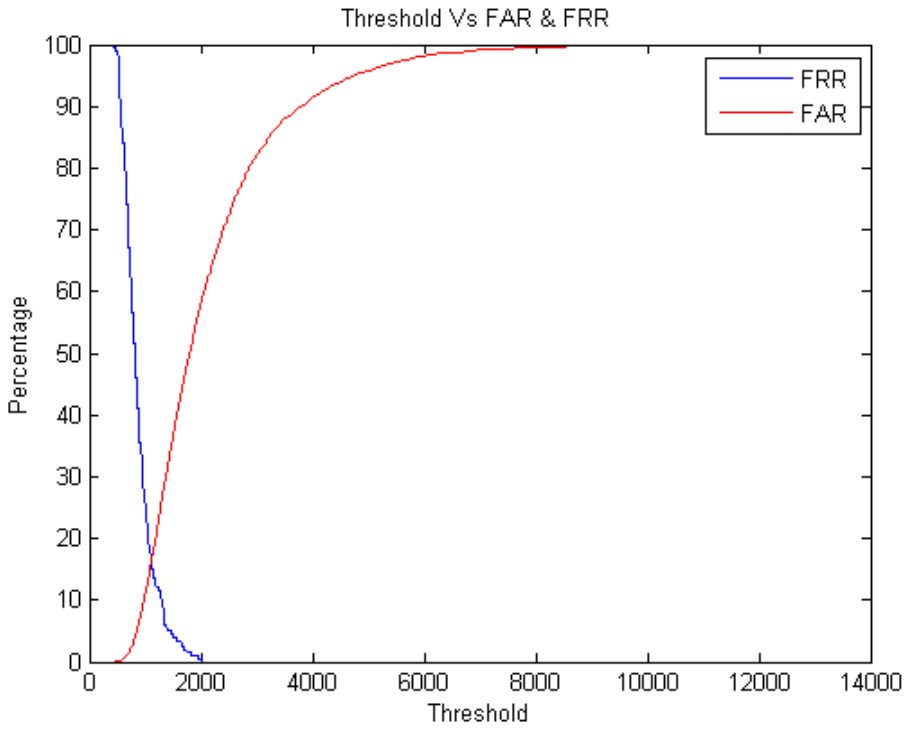

Figure 5.25: EER of 480 Zernike features tested on DB1

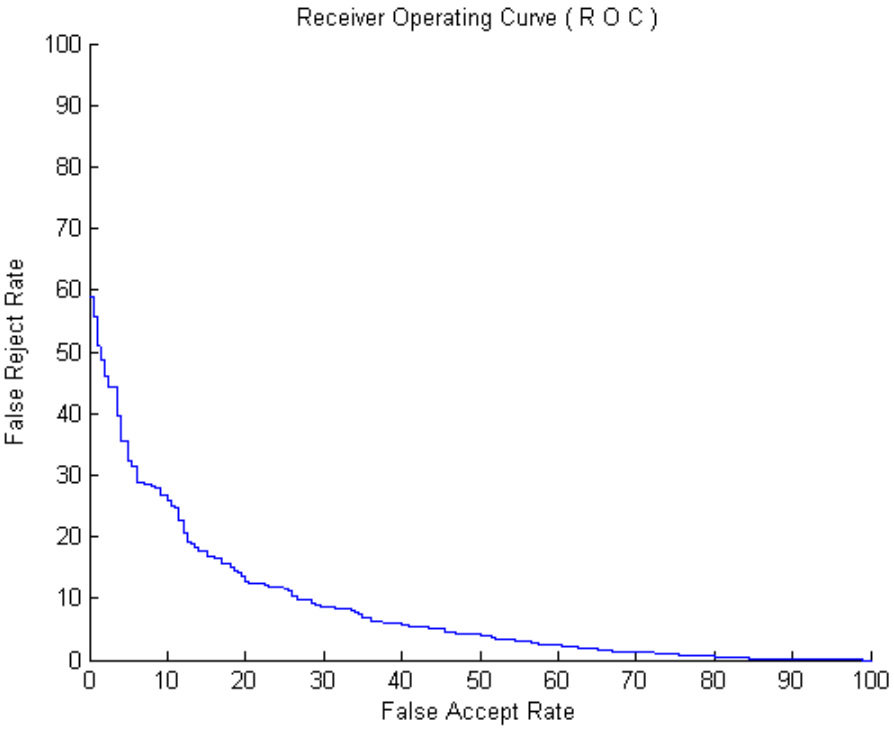

Figure 5.26: ROC Curve of 480 Zernike features tested on DB1

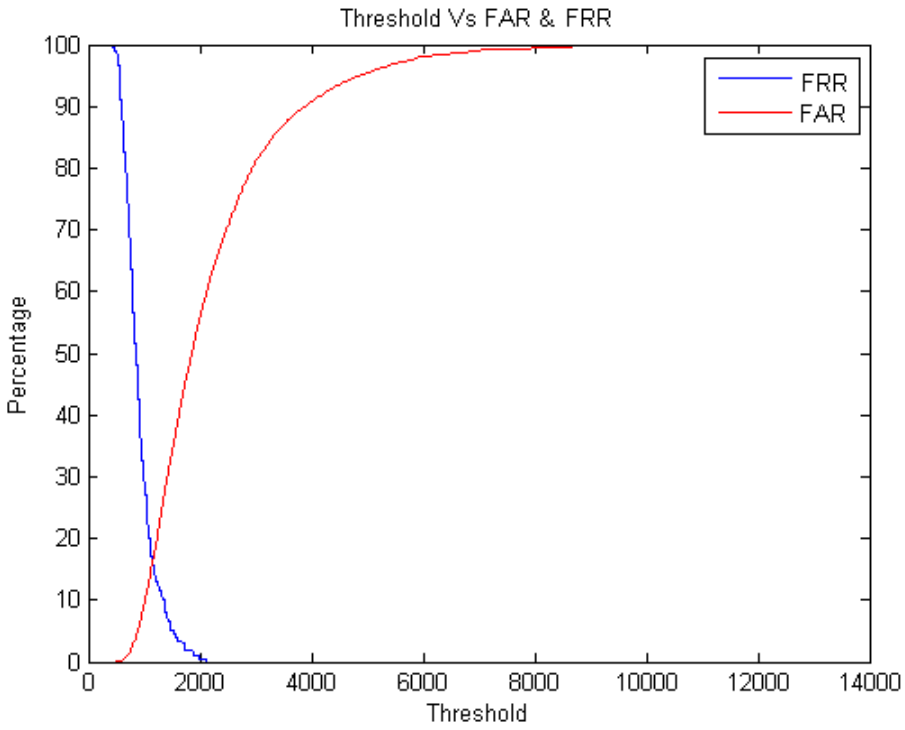

Figure 5.27: EER of 500 Zernike features tested on DB1

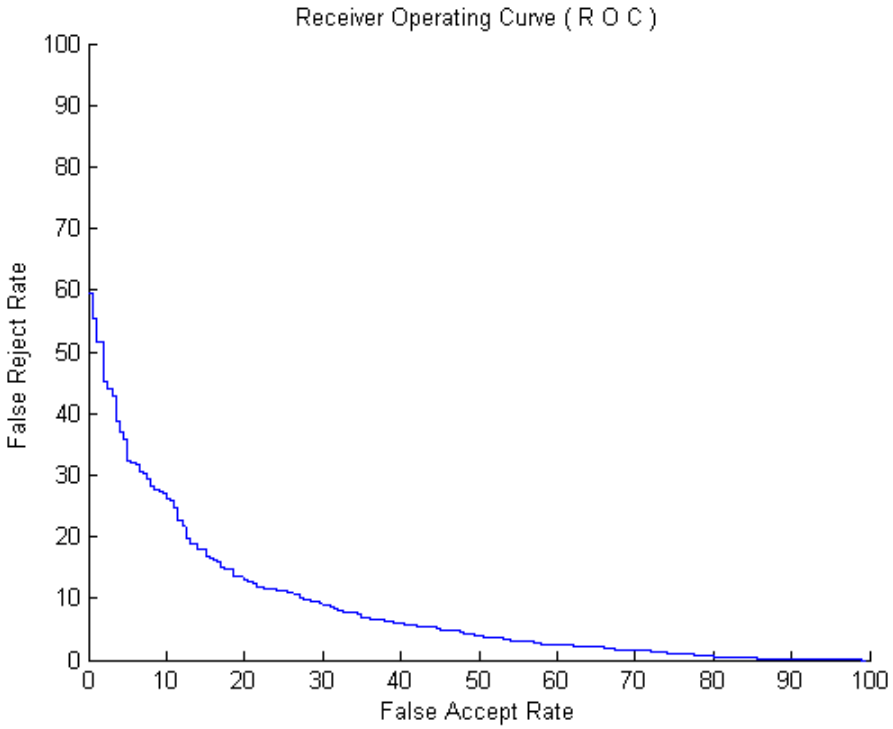

Figure 5.28: ROC Curve of 500 Zernike features tested on DB1

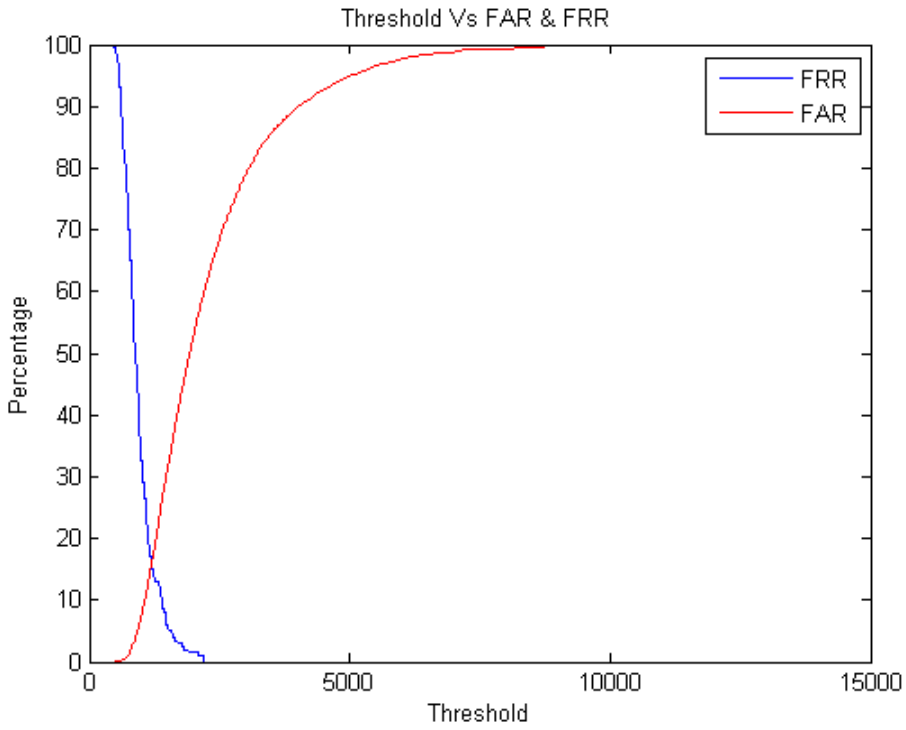

Figure 5.29: EER of 520 Zernike features tested on DB1

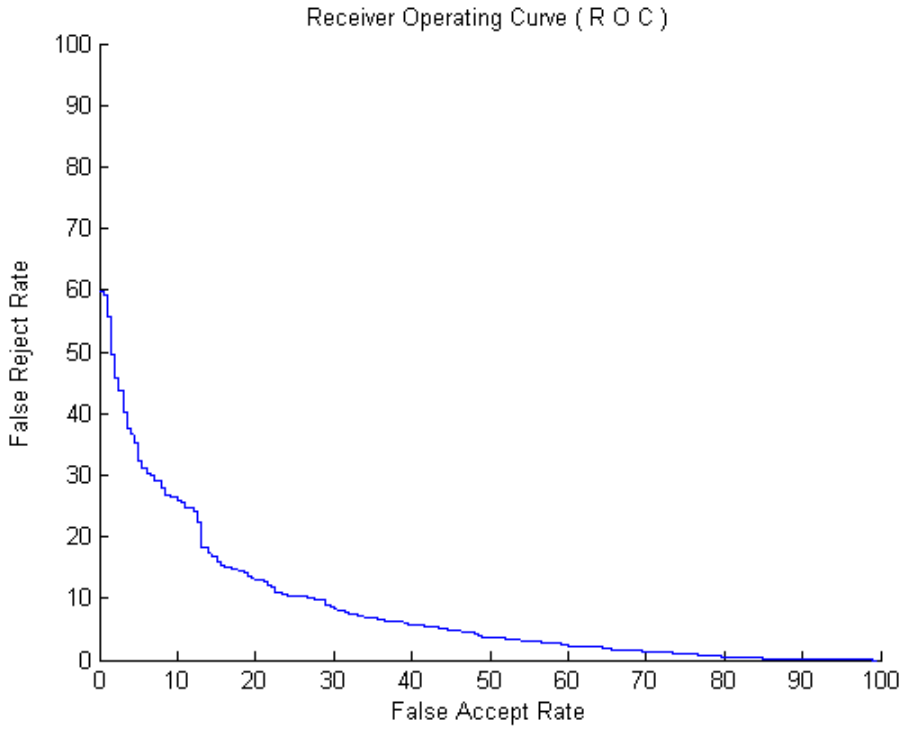

Figure 5.30: ROC Curve of 520 Zernike features tested on DB1

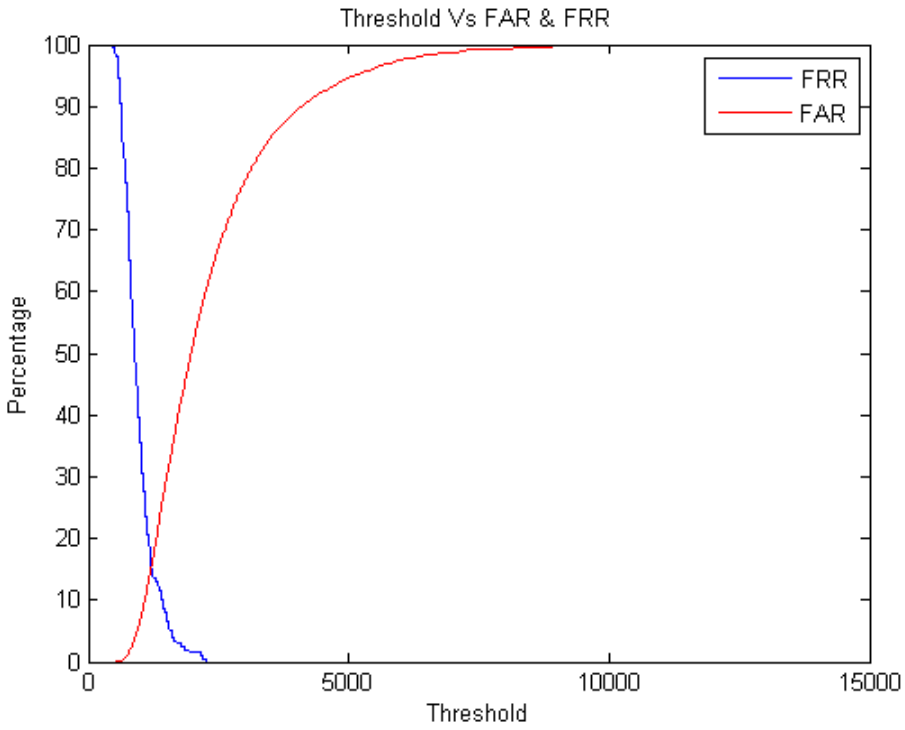

Figure 5.31: EER of 530 Zernike features tested on DB1

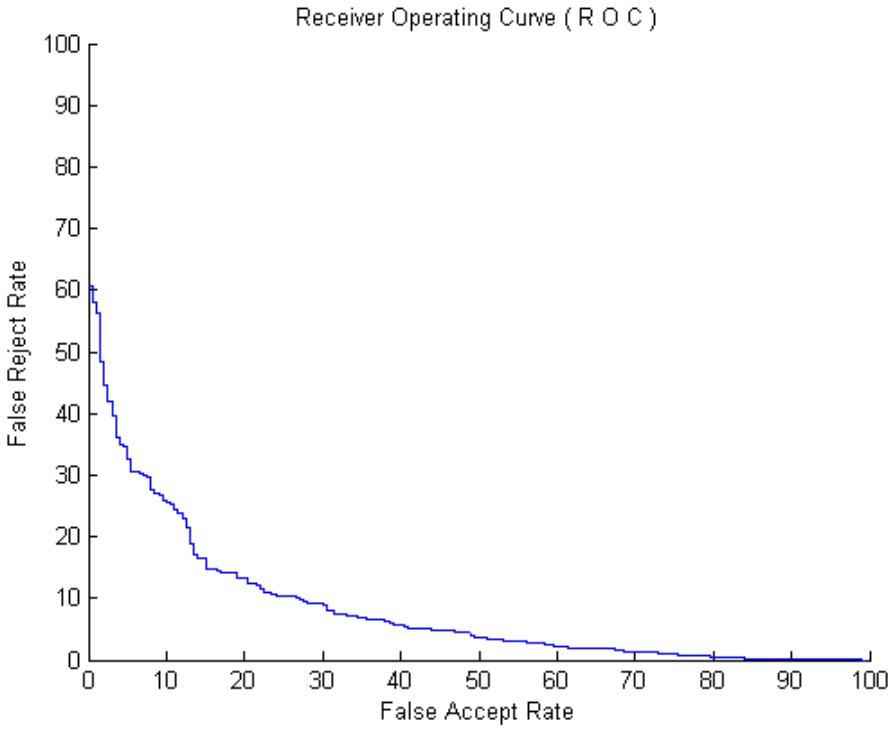

Figure 5.32: ROC Curve of 530 Zernike features tested on DB1

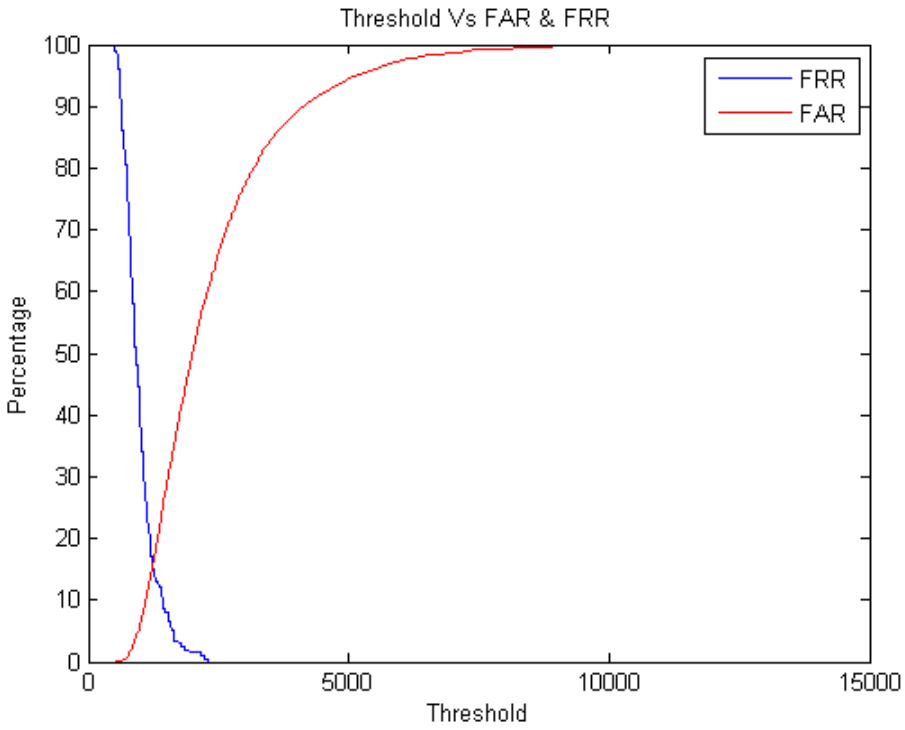

Figure 5.33: EER of 540 Zernike features tested on DB1

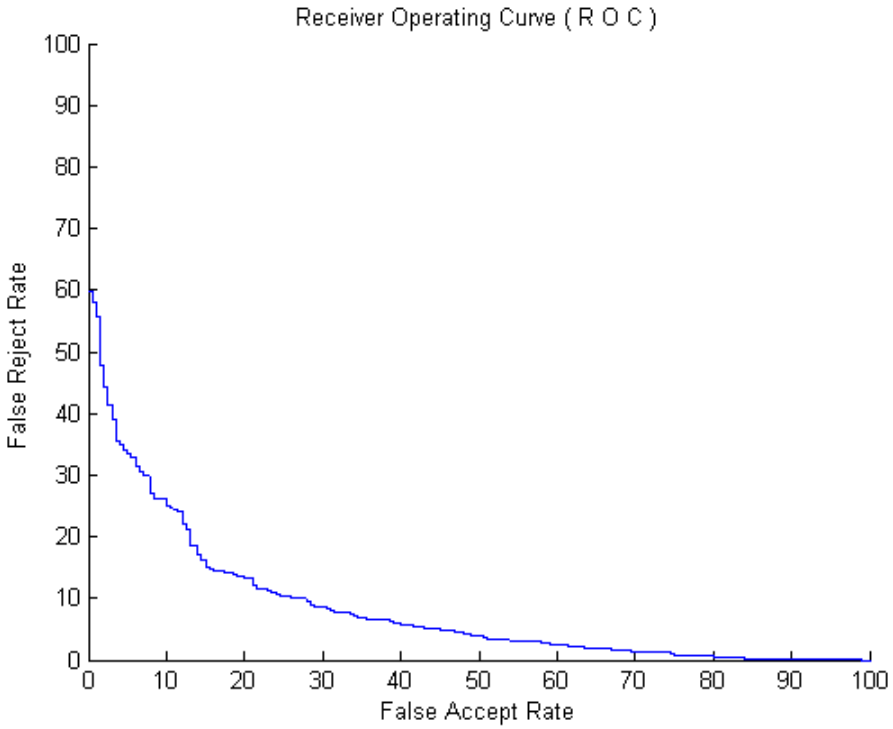

Figure 5.34: ROC Curve of 540 Zernike features tested on DB1

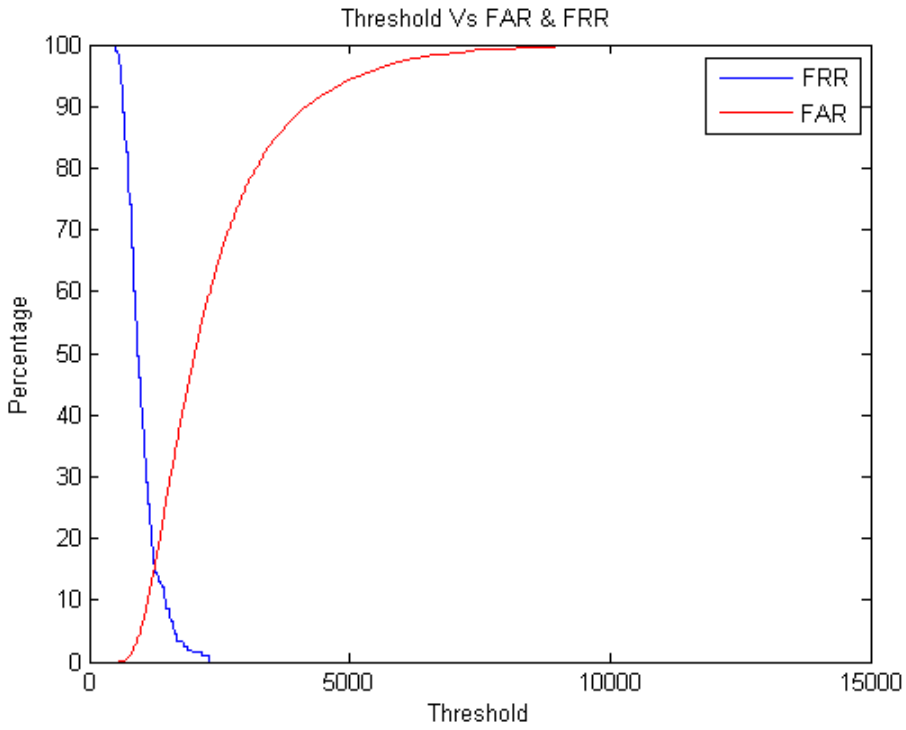

Figure 5.35: EER of 550 Zernike features tested on DB1

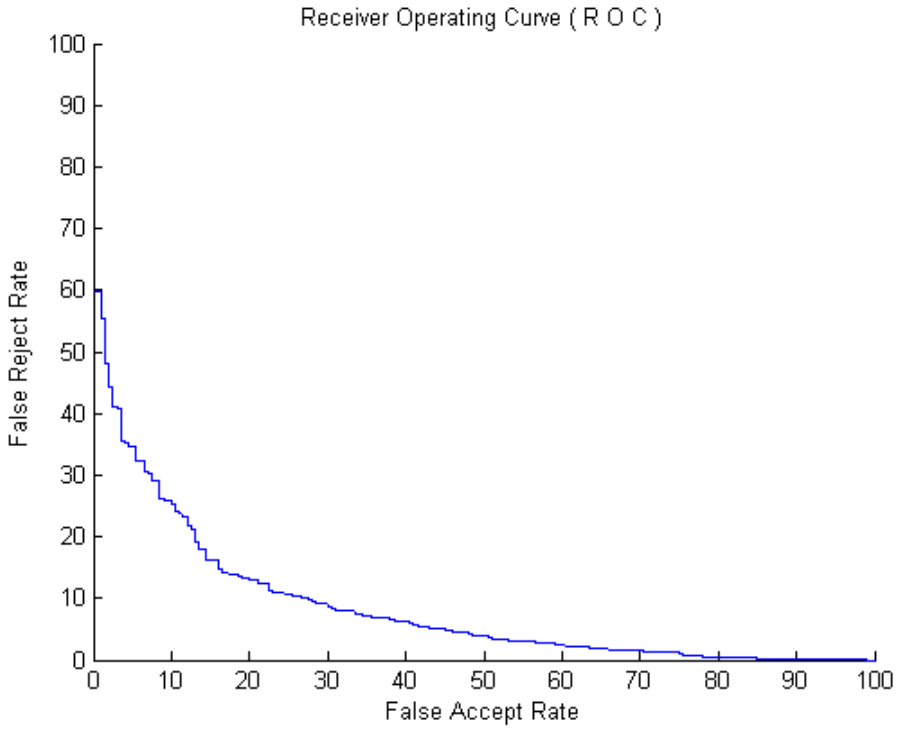

Figure 5.36: ROC Curve of 550 Zernike features tested on DB1

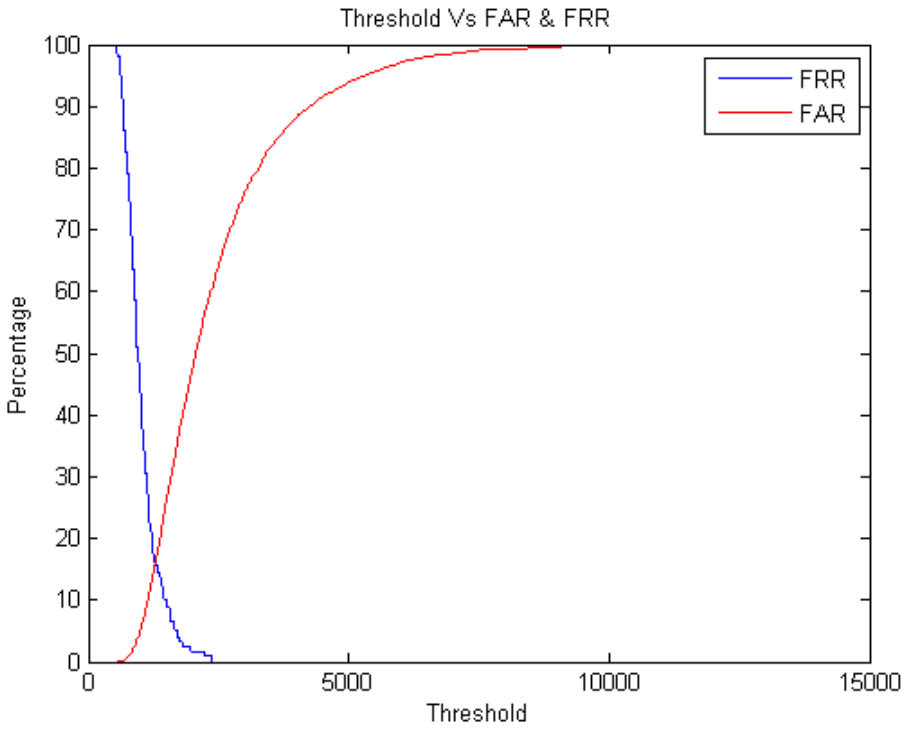

Figure 5.37: EER of 560 Zernike features tested on DB1

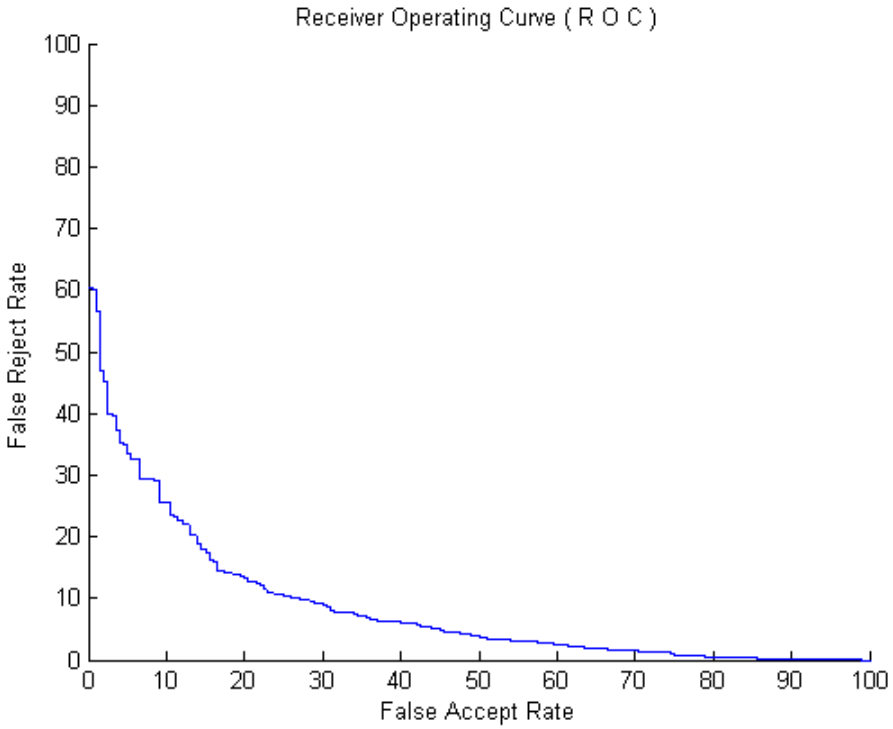

Figure 5.38: ROC Curve of 560 Zernike features tested on DB1

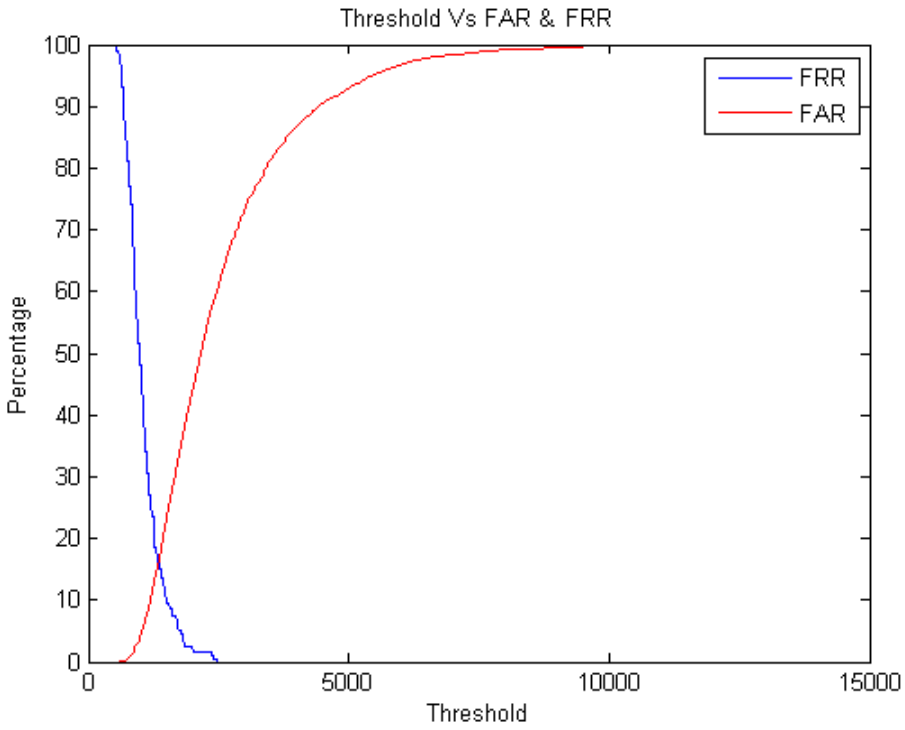

Figure 5.39: EER of 580 Zernike features tested on DB1

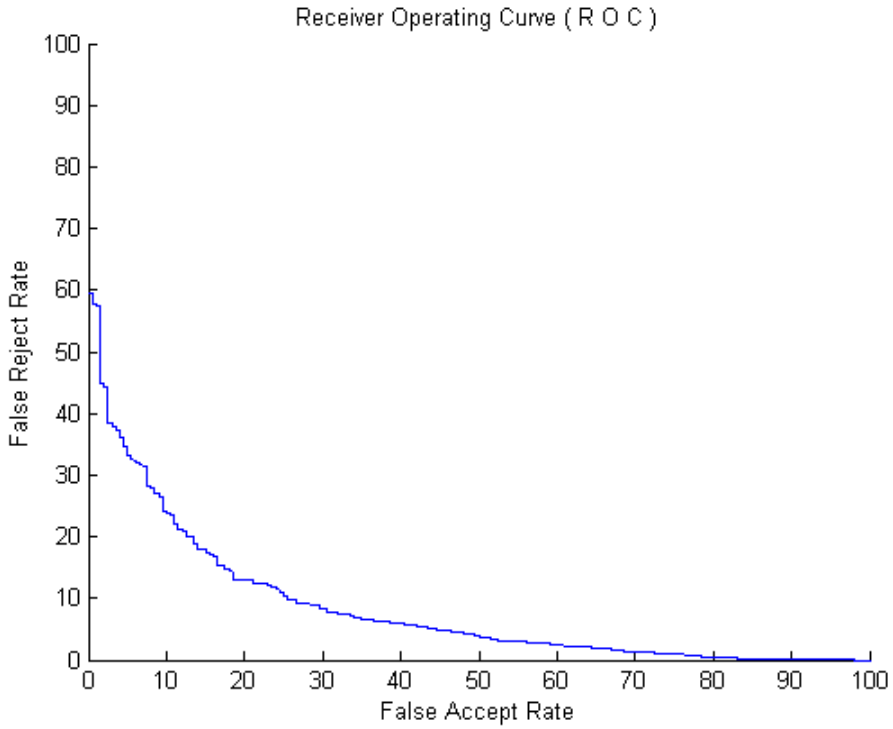

Figure 5.40: ROC Curve of 580 Zernike features tested on DB1

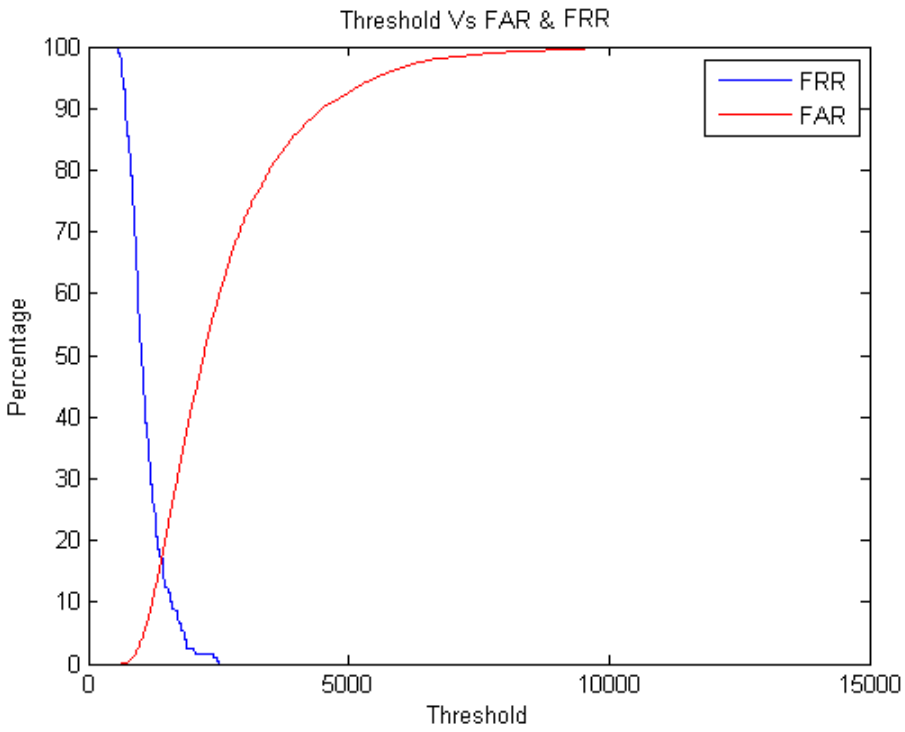

Figure 5.41: EER of 600 Zernike features tested on DB1

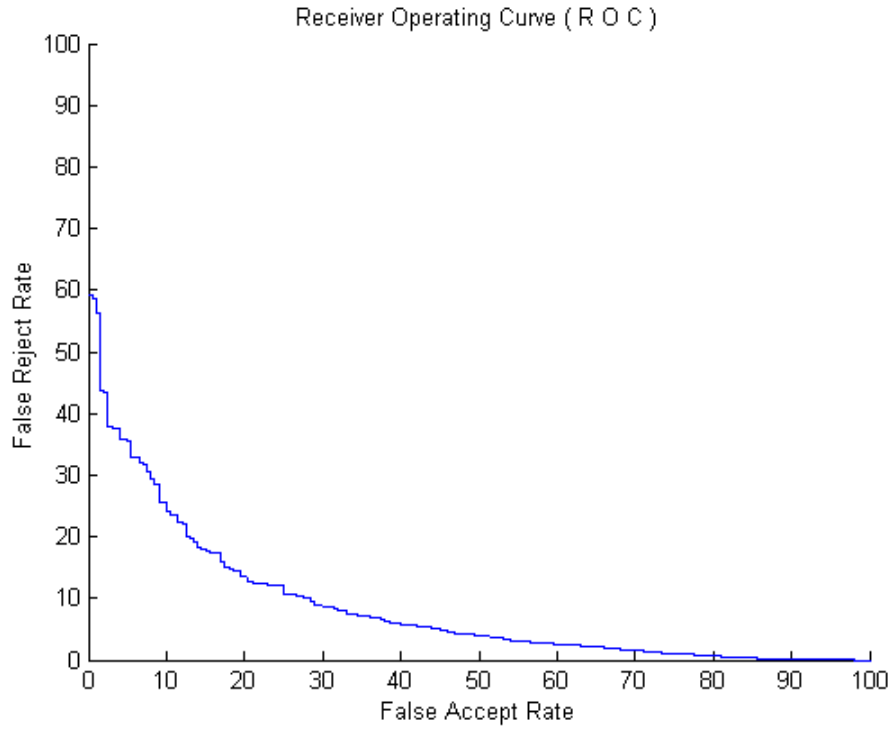

Figure 5.42: ROC Curve of 600 Zernike features tested on DB1

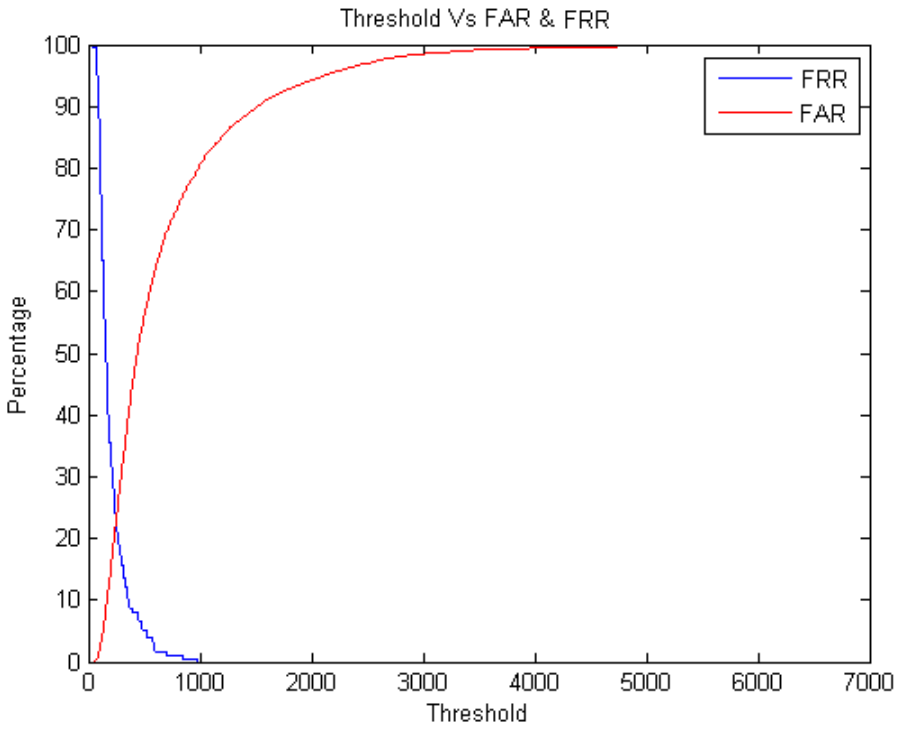

Figure 5.43: EER of 100 Zernike features tested on DB2

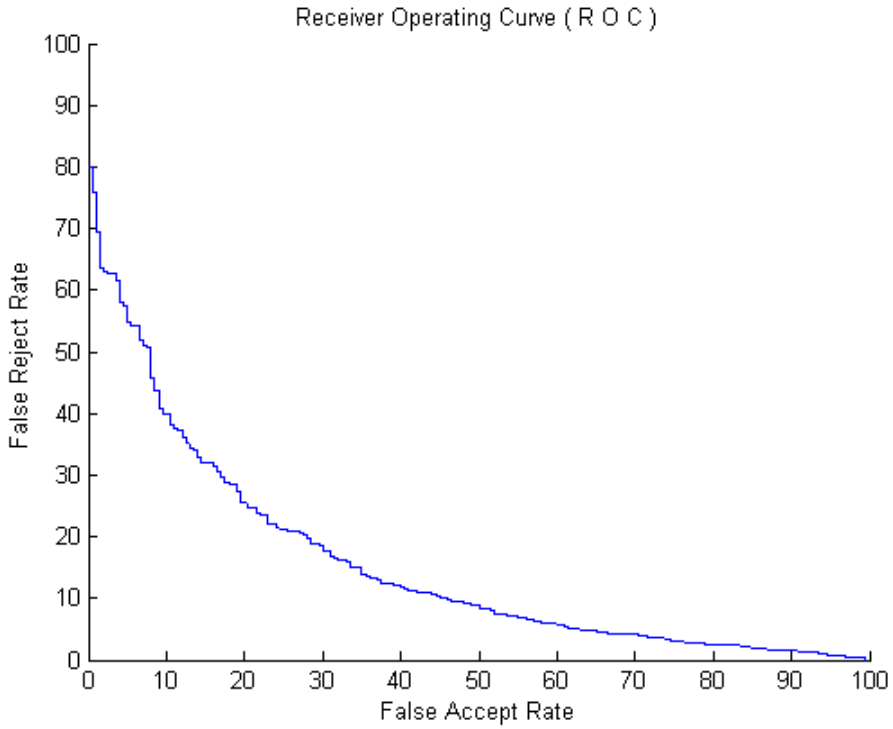

Figure 5.44: ROC Curve of 100 Zernike features tested on DB2

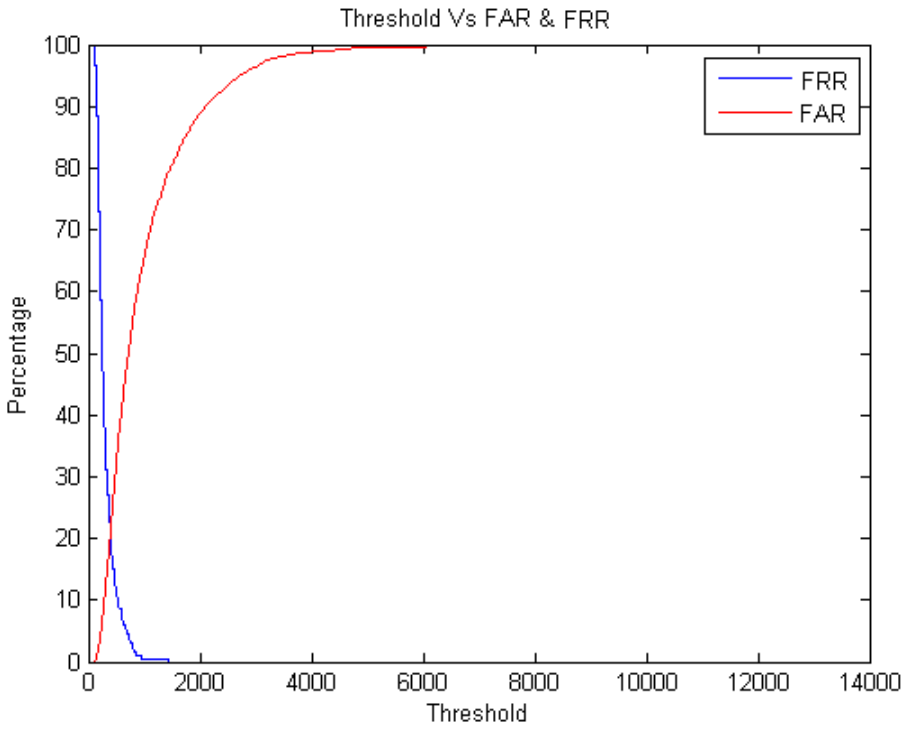

Figure 5.45: EER of 150 Zernike features tested on DB2

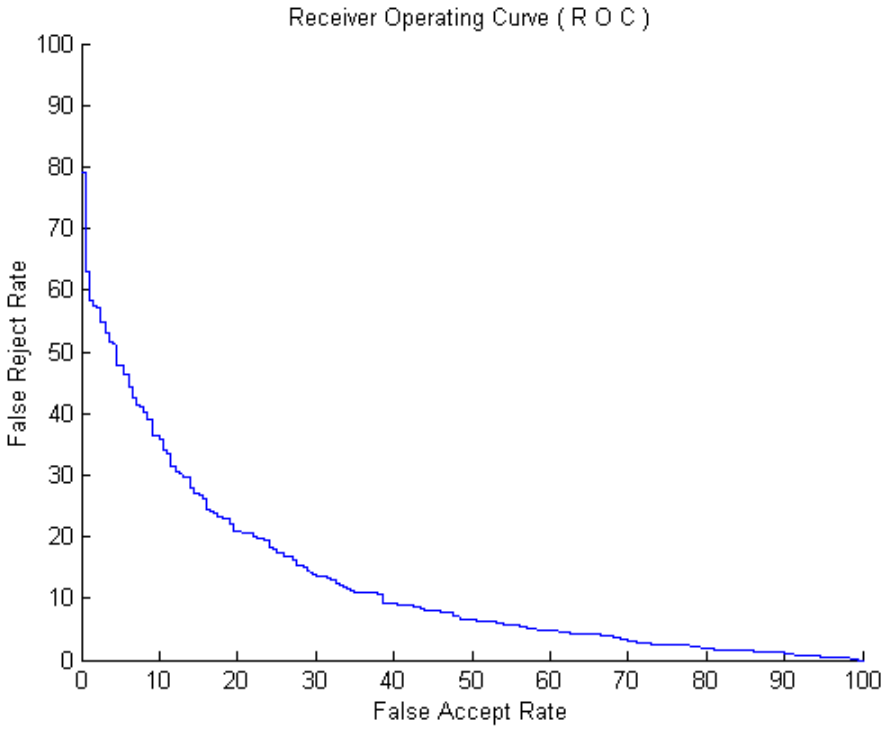

Figure 5.46: ROC Curve of 150 Zernike features tested on DB2

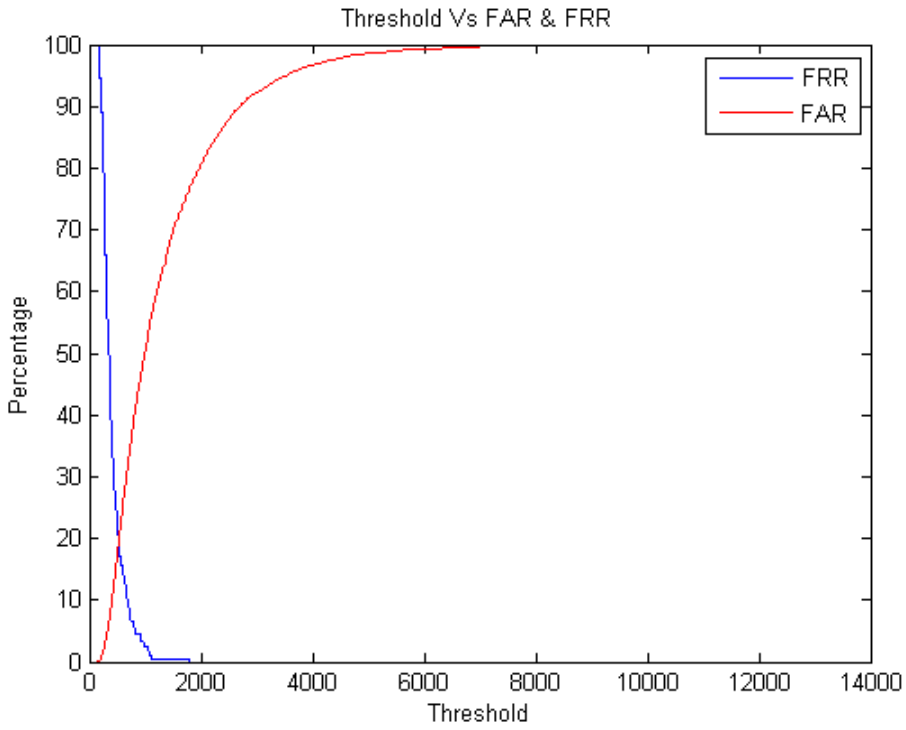

Figure 5.47: EER of 200 Zernike features tested on DB2

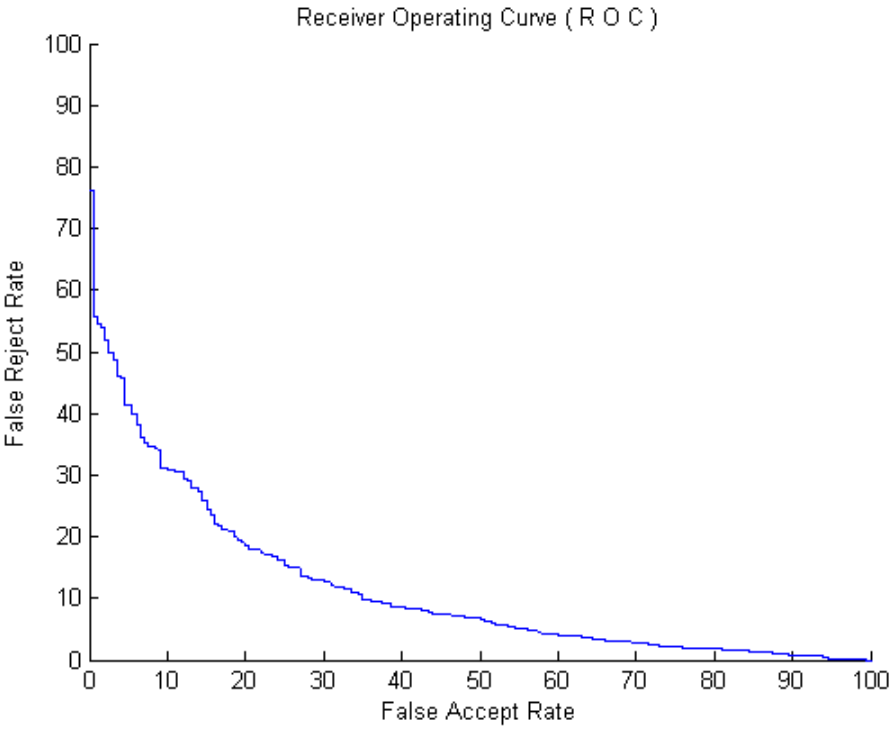

Figure 5.48: ROC Curve of 200 Zernike features tested on DB2

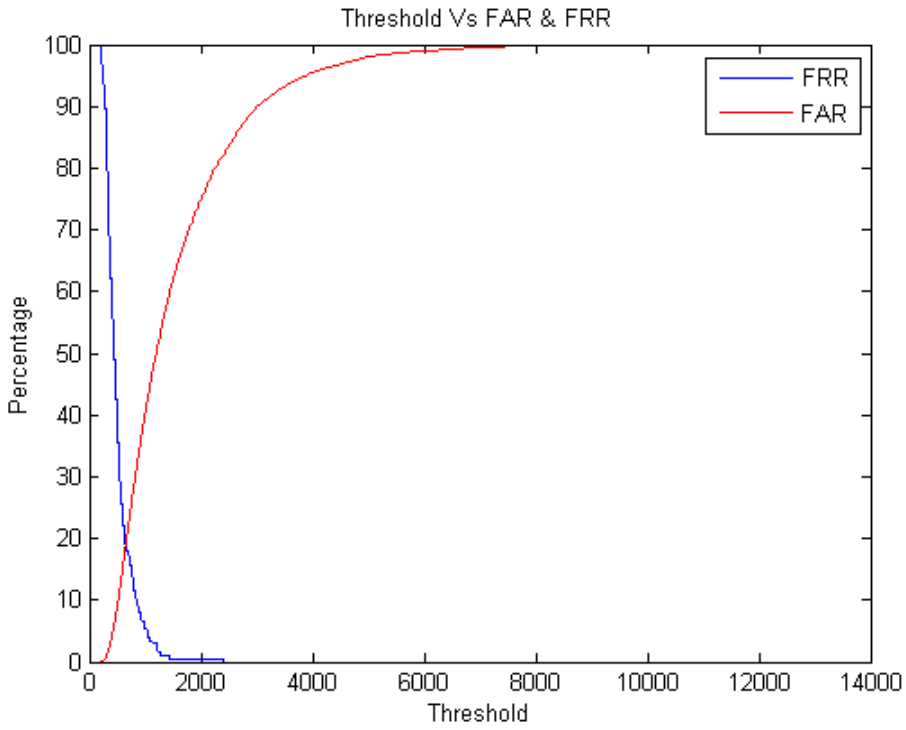

Figure 5.49: EER of 250 Zernike features tested on DB2

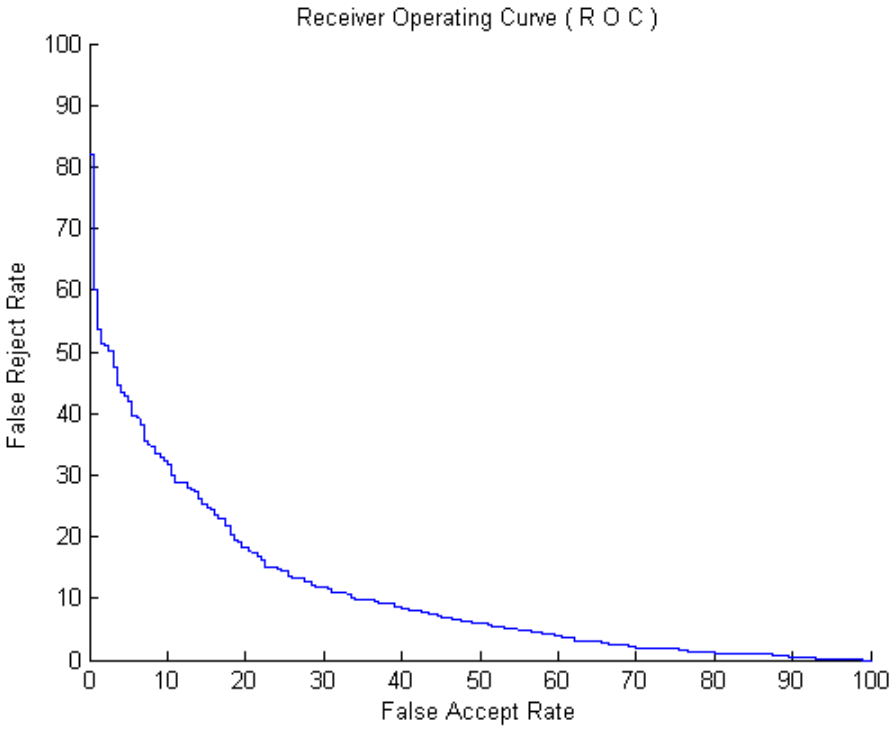

Figure 5.50: ROC Curve of 250 Zernike features tested on DB2

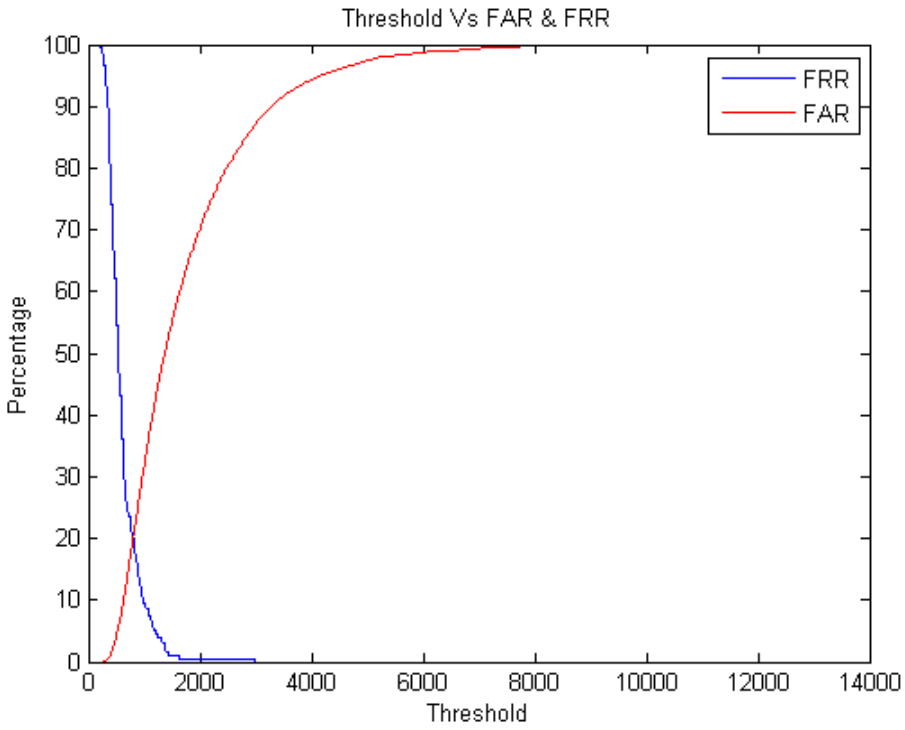

Figure 5.51: EER of 300 Zernike features tested on DB2

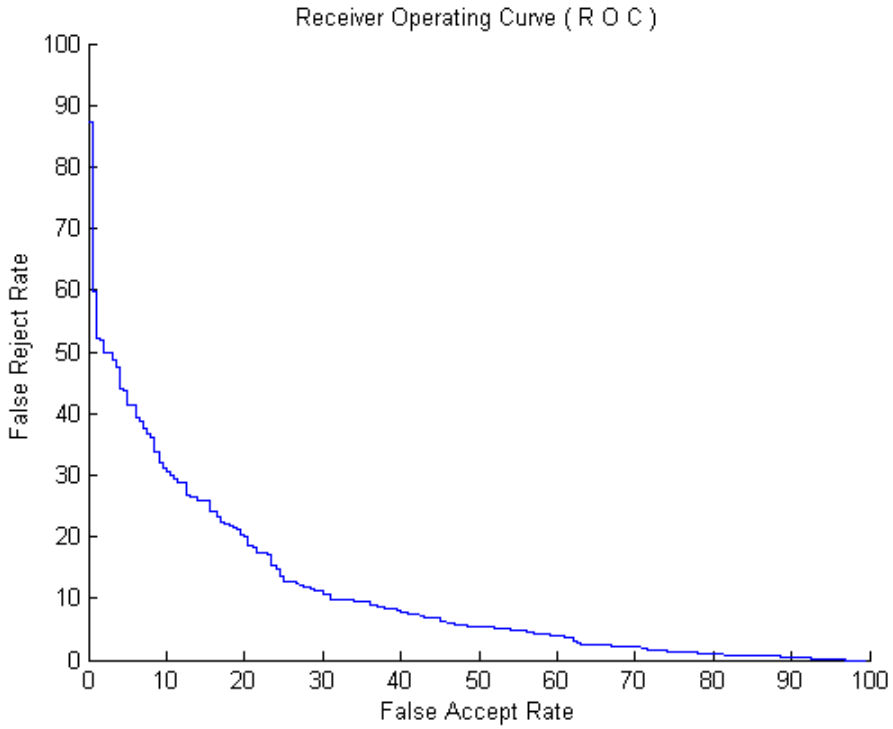

Figure 5.52: ROC Curve of 300 Zernike features tested on DB2

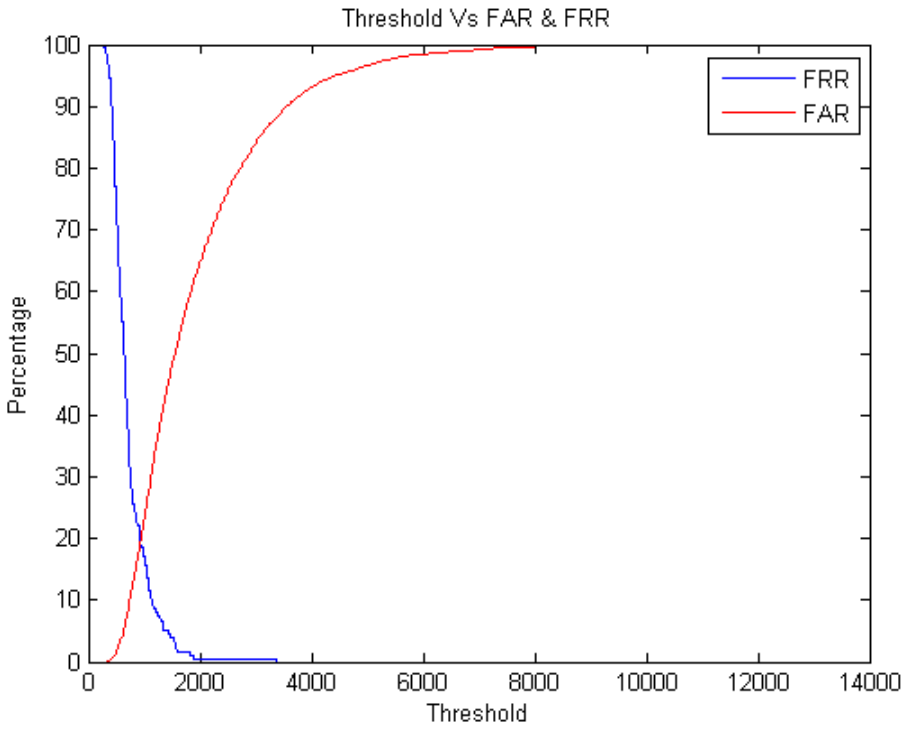

Figure 5.53: EER of 350 Zernike features tested on DB2

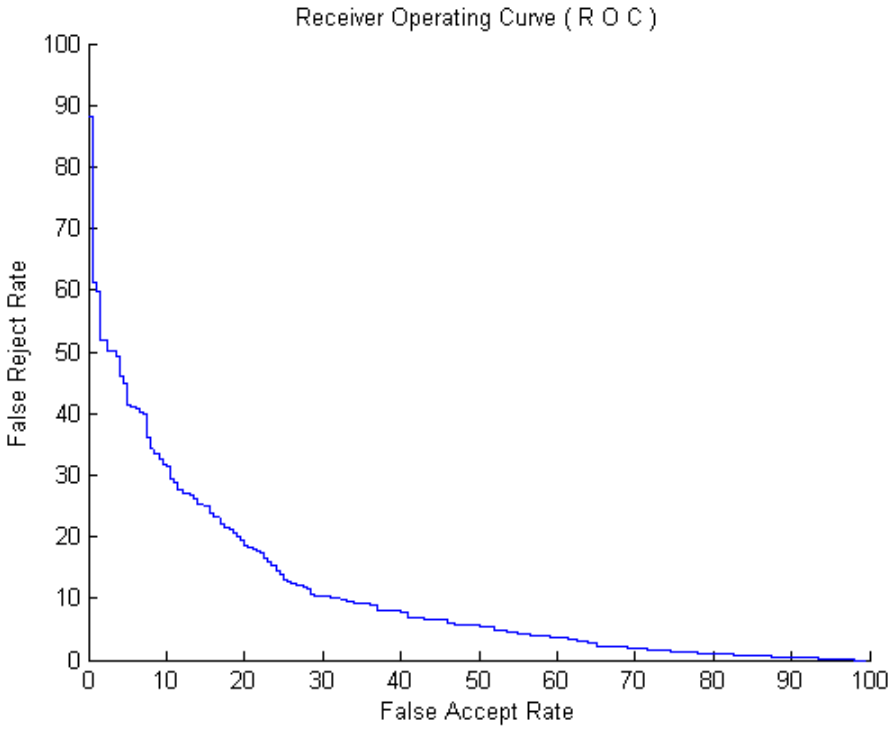

Figure 5.54: ROC Curve of 350 Zernike features tested on DB2

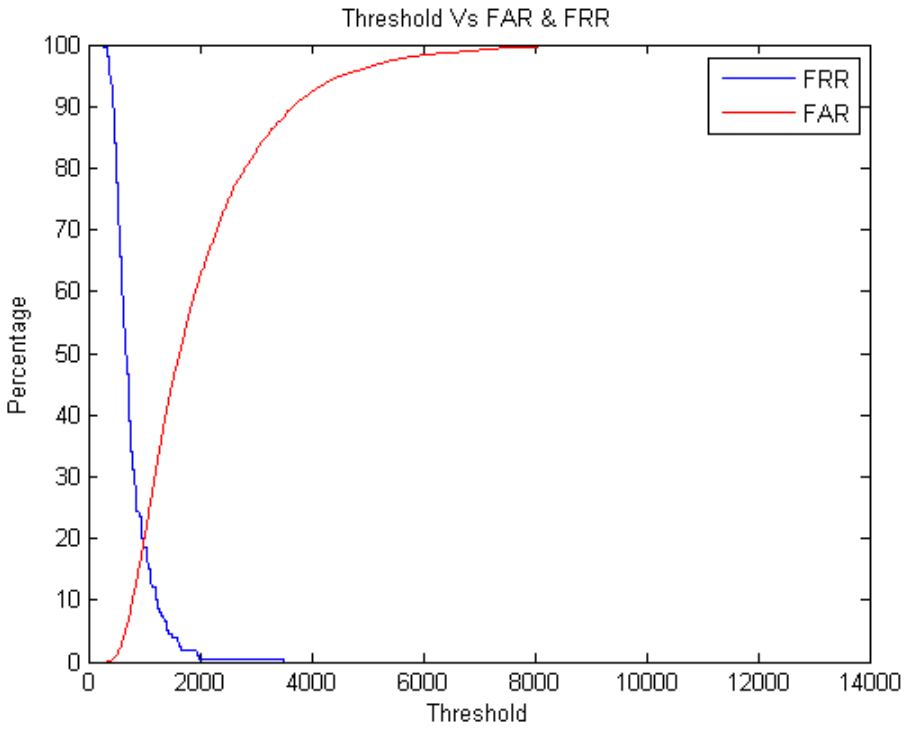

Figure 5.55: EER of 370 Zernike features tested on DB2

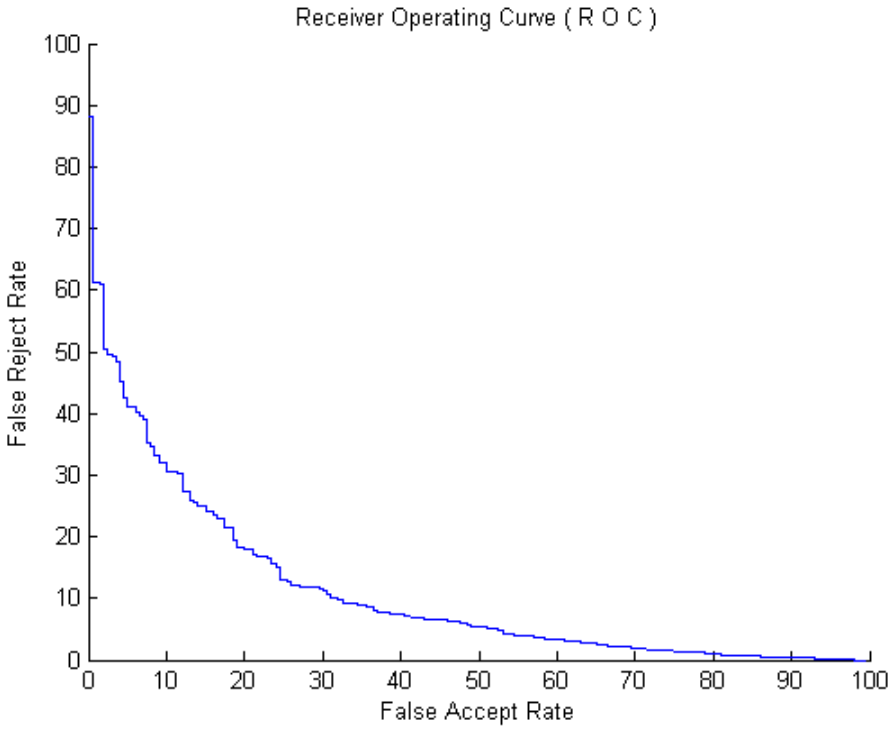

Figure 5.56: ROC Curve of 370 Zernike features tested on DB2

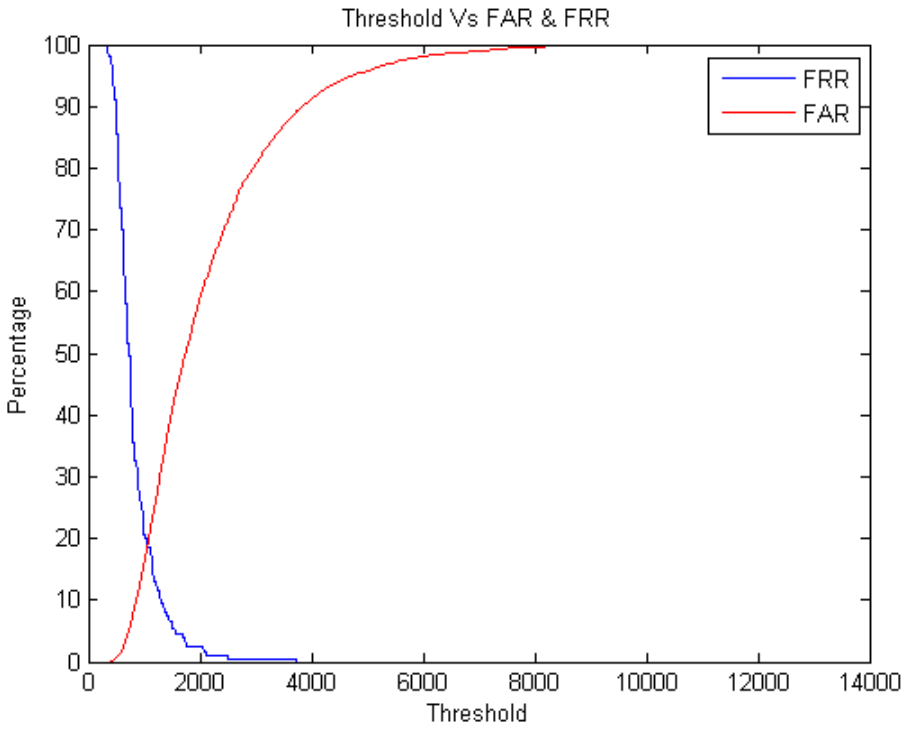

Figure 5.57: EER of 398 Zernike features tested on DB2

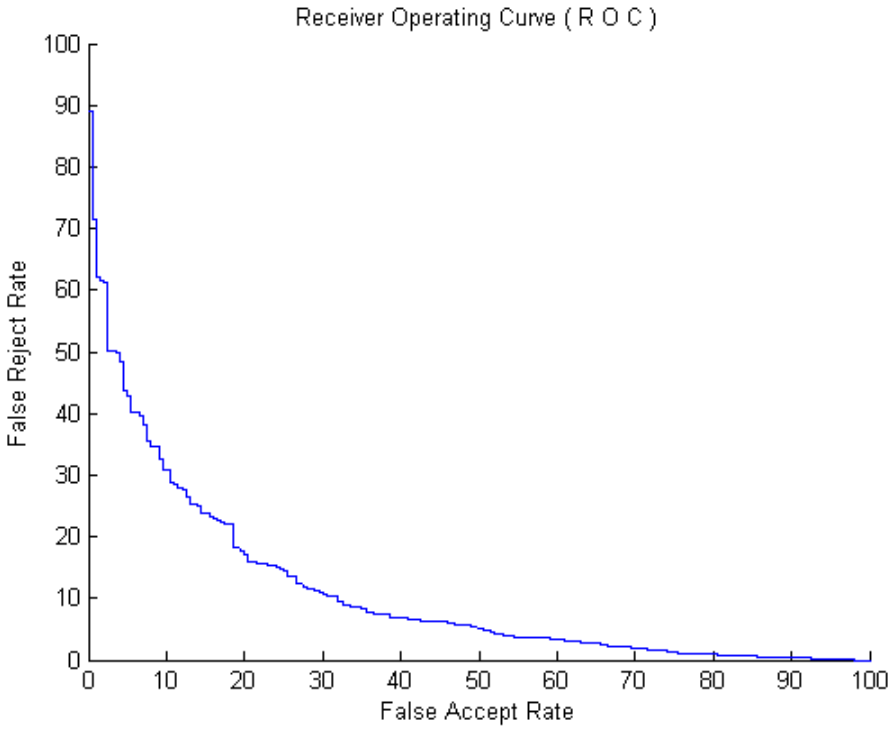

Figure 5.58: ROC Curve of 398 Zernike features tested on DB2

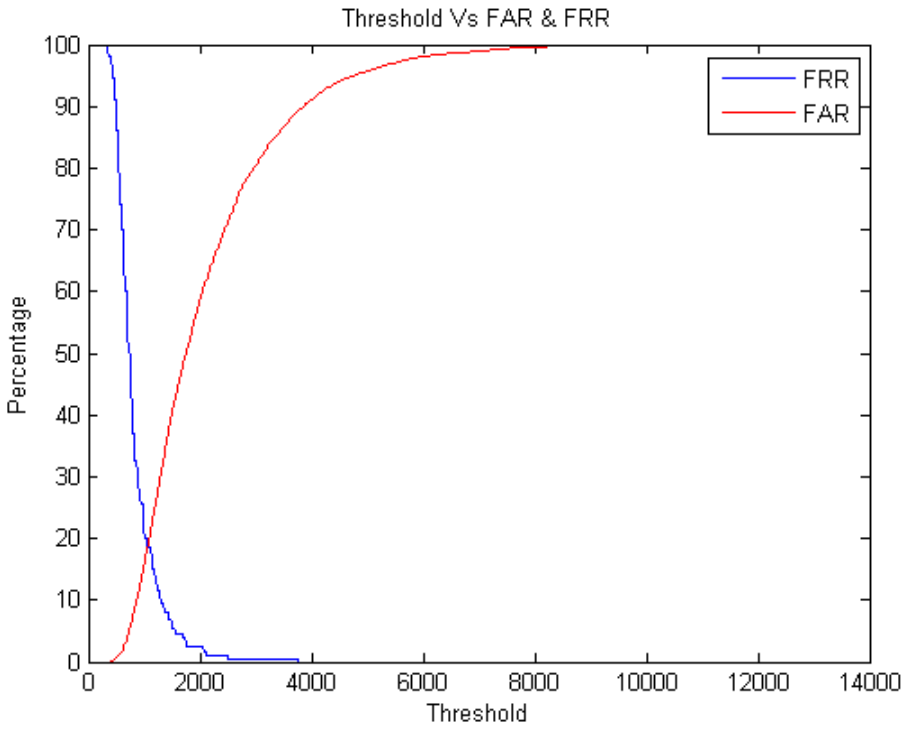

Figure 5.59: EER of 400 Zernike features tested on DB2

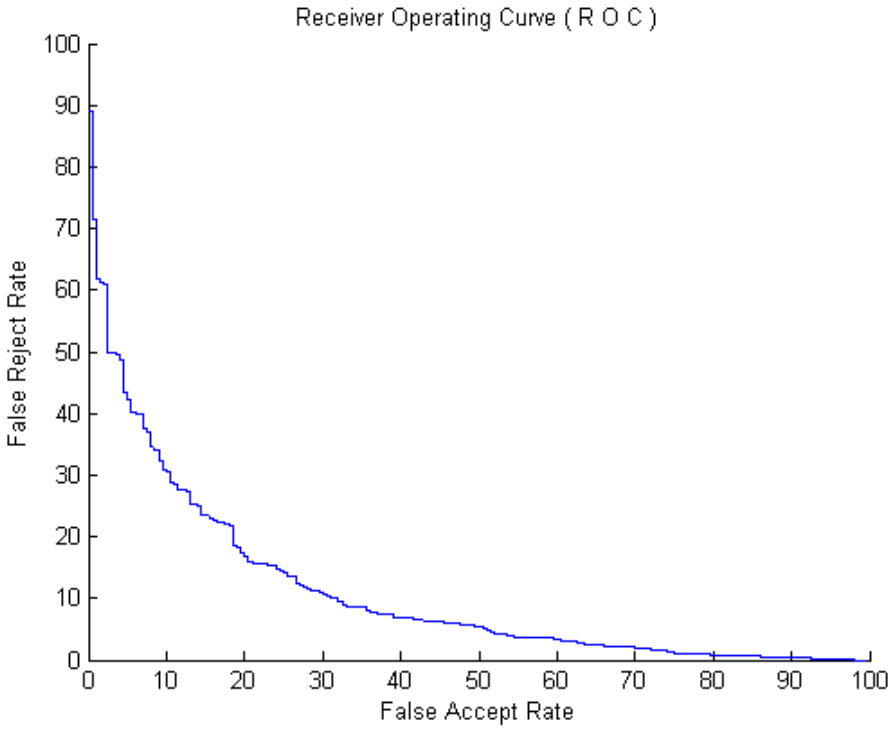

Figure 5.60: ROC Curve of 400 Zernike features tested on DB2

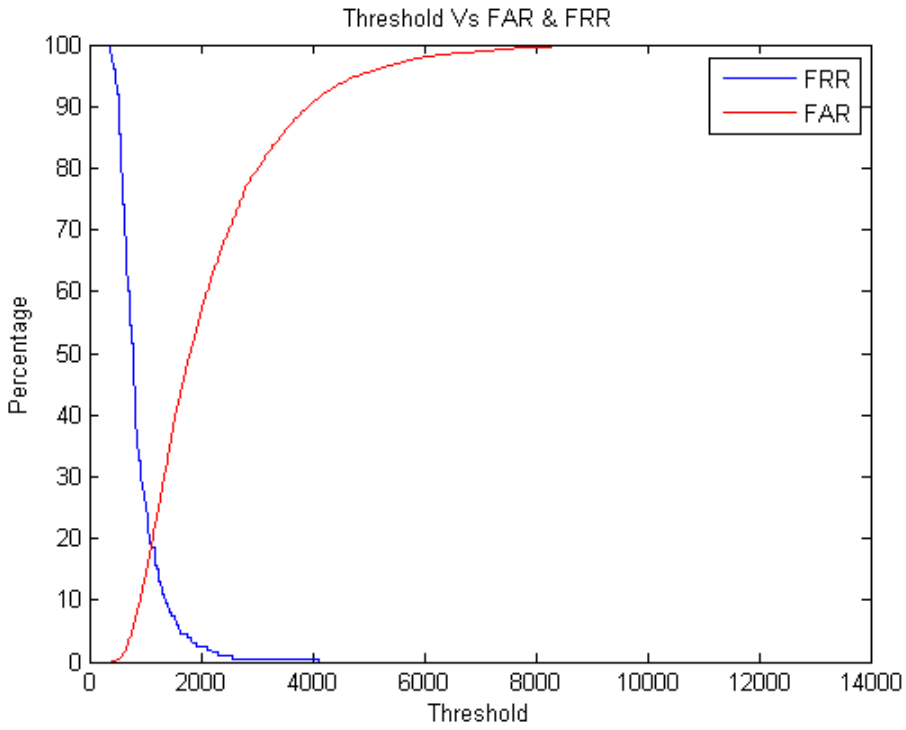

Figure 5.61: EER of 420 Zernike features tested on DB2

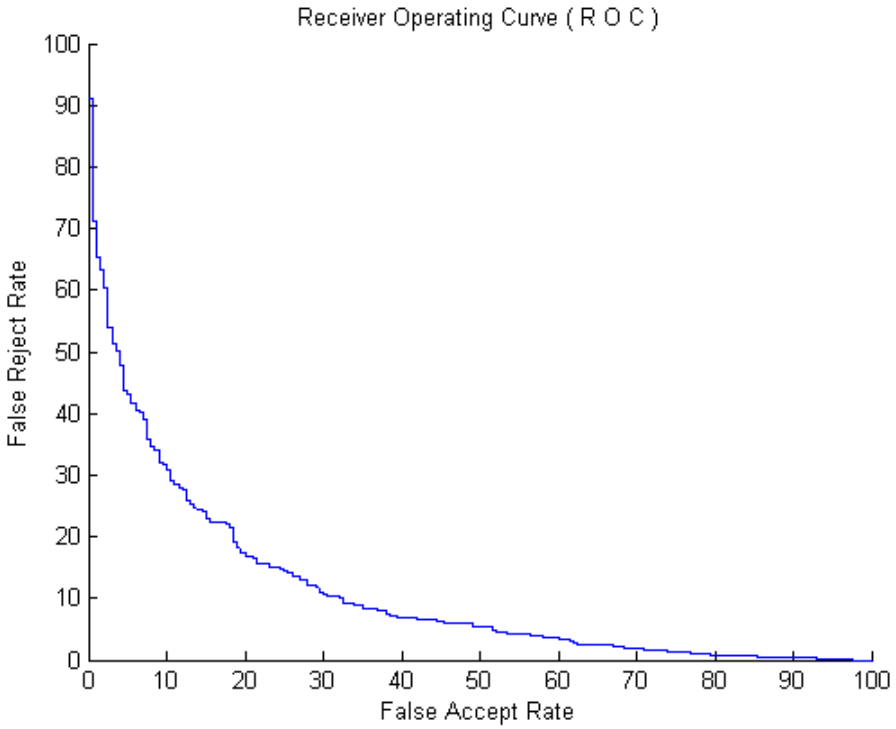

Figure 5.62: ROC Curve of 420 Zernike features tested on DB2

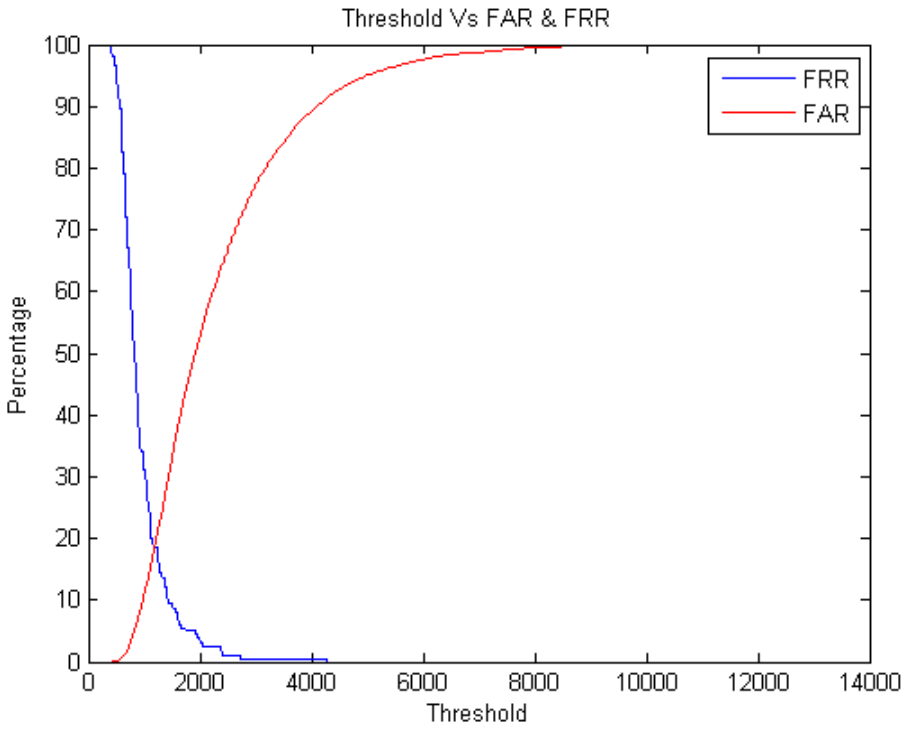

Figure 5.63: EER of 450 Zernike features tested on DB2

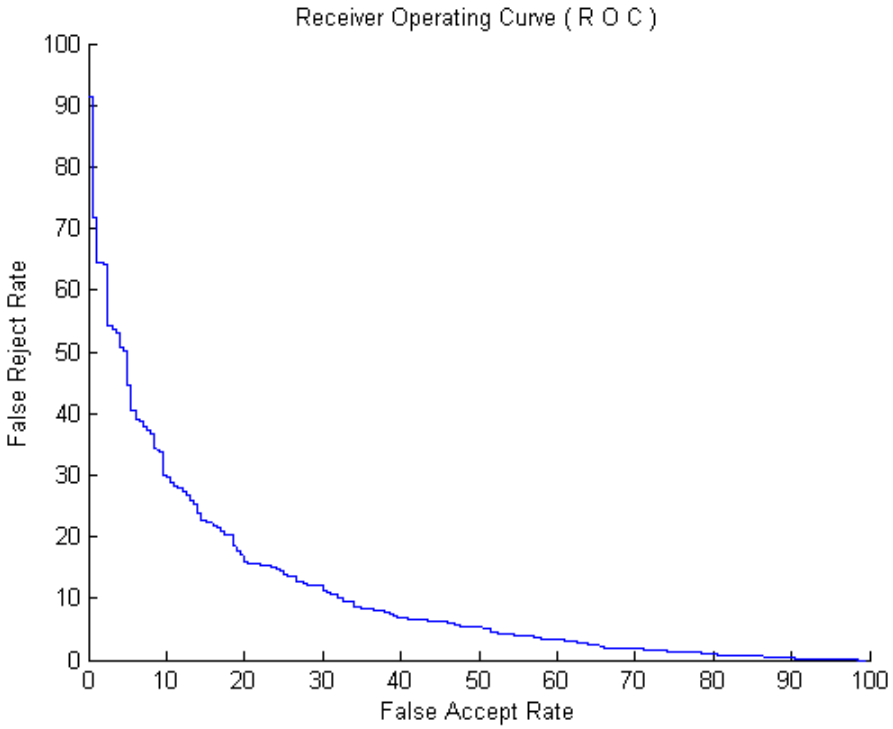

Figure 5.64: ROC Curve of 450 Zernike features tested on DB2

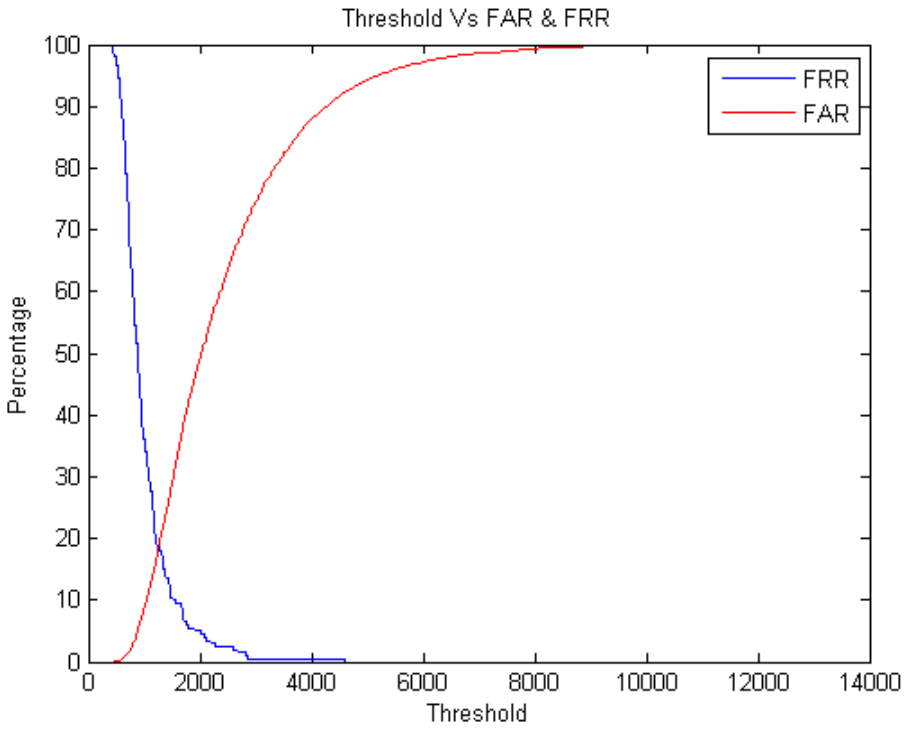

Figure 5.65: EER of 480 Zernike features tested on DB2

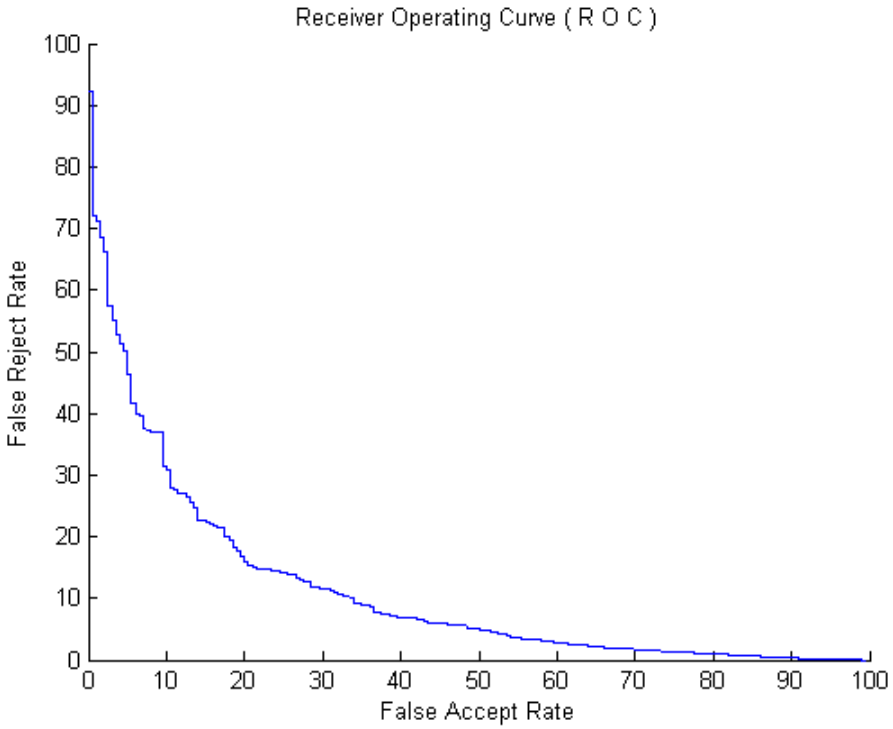

Figure 5.66: ROC Curve of 480 Zernike features tested on DB2

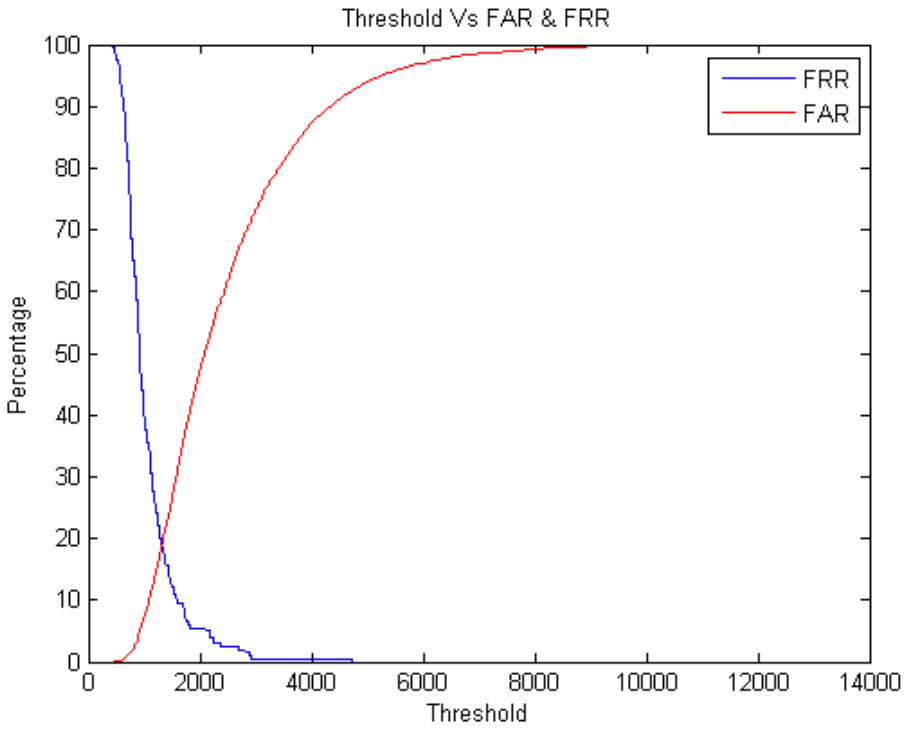

Figure 5.67: EER of 500 Zernike features tested on DB2

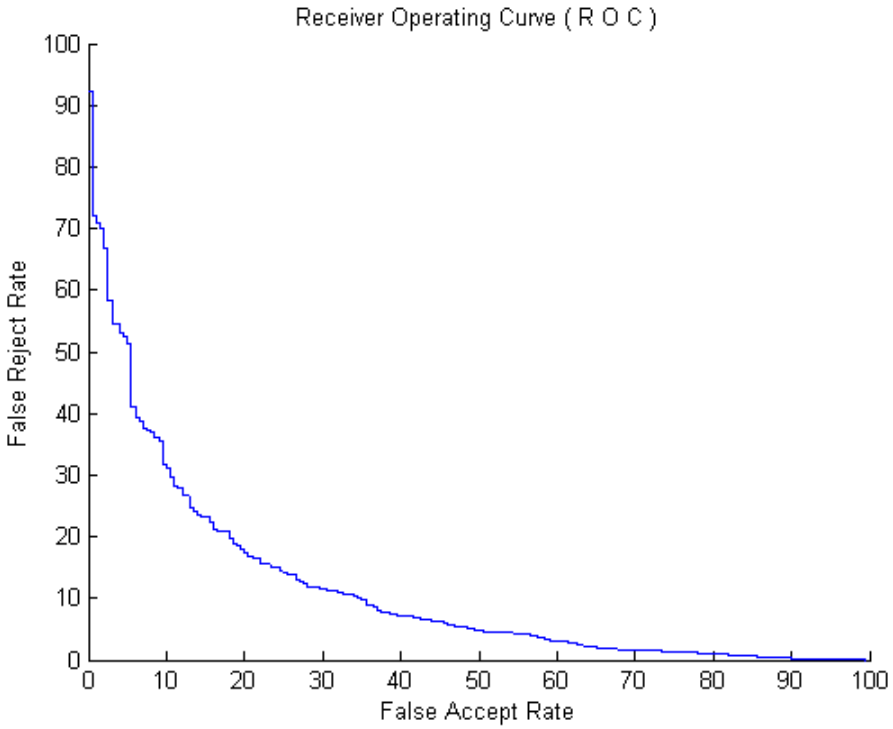

Figure 5.68: ROC Curve of 500 Zernike features tested on DB2

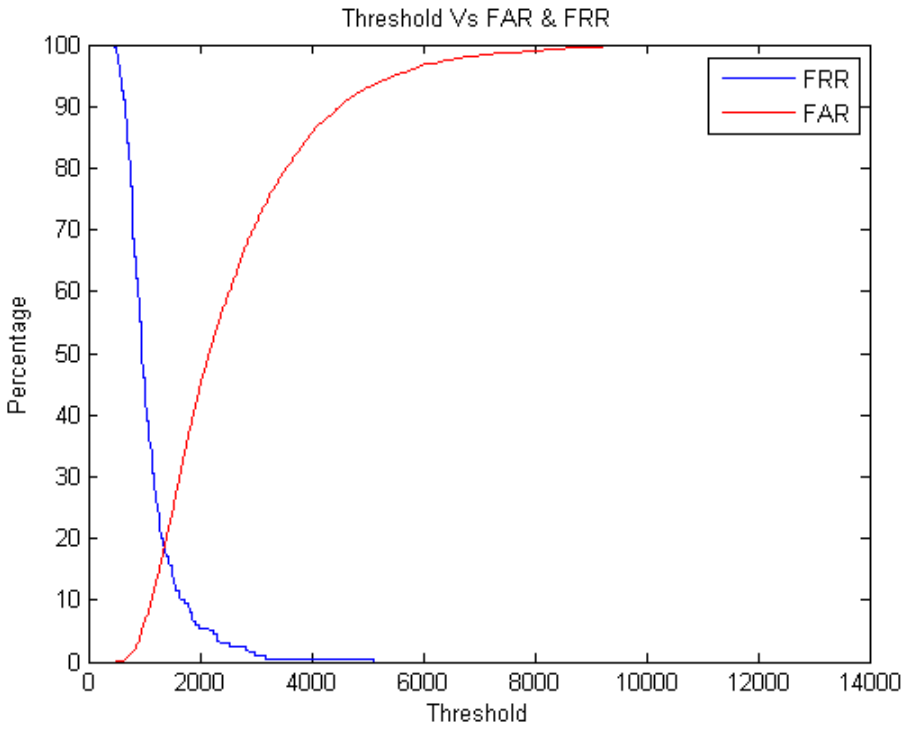

Figure 5.69: EER of 520 Zernike features tested on DB2

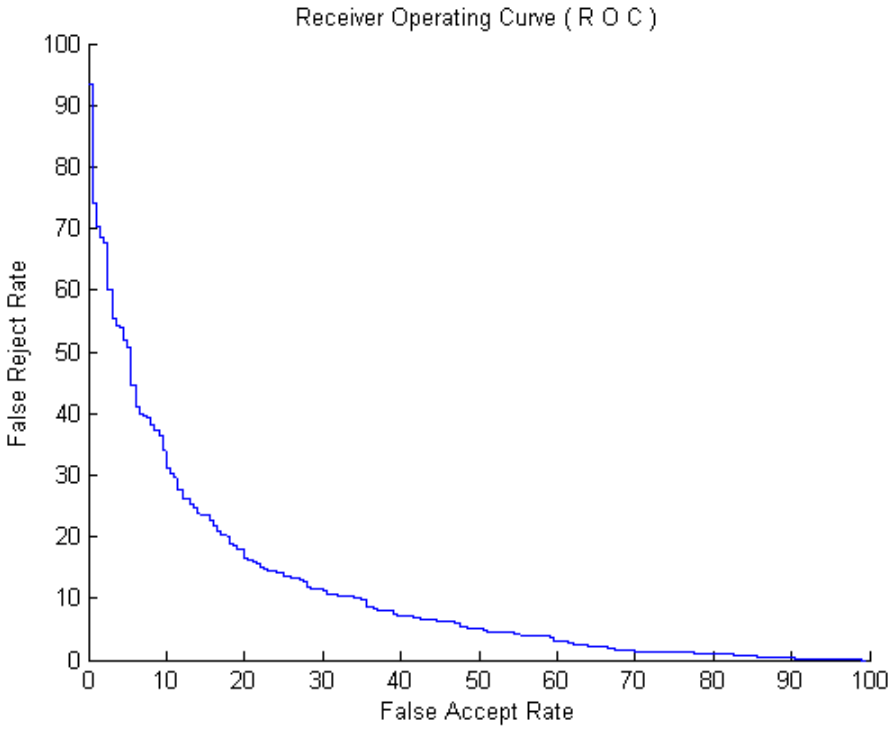

Figure 5.70: ROC Curve of 520 Zernike features tested on DB2

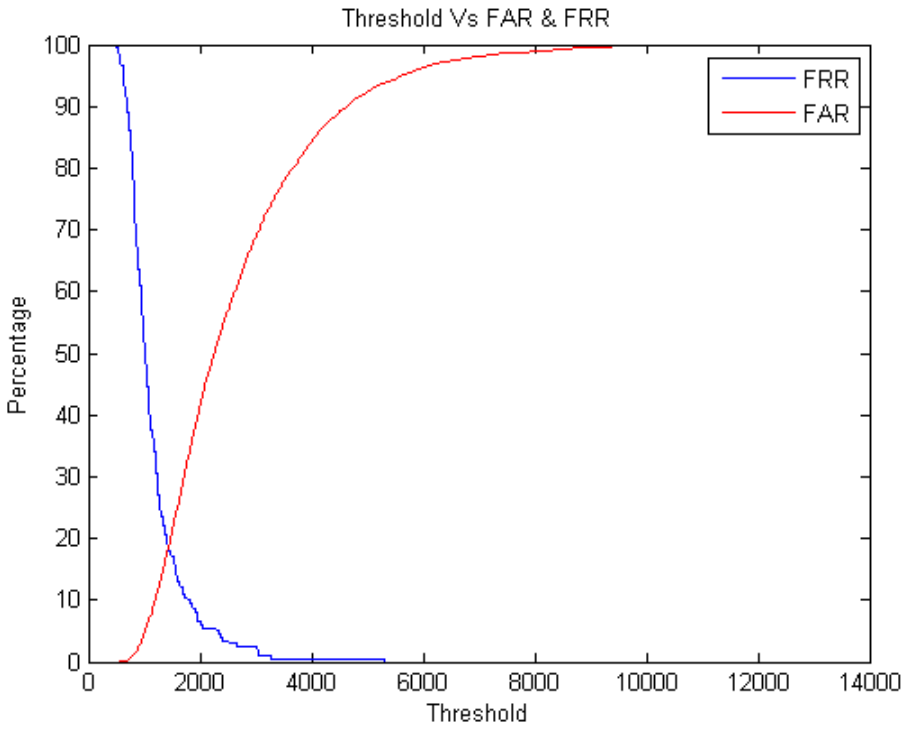

Figure 5.71: EER of 550 Zernike features tested on DB2

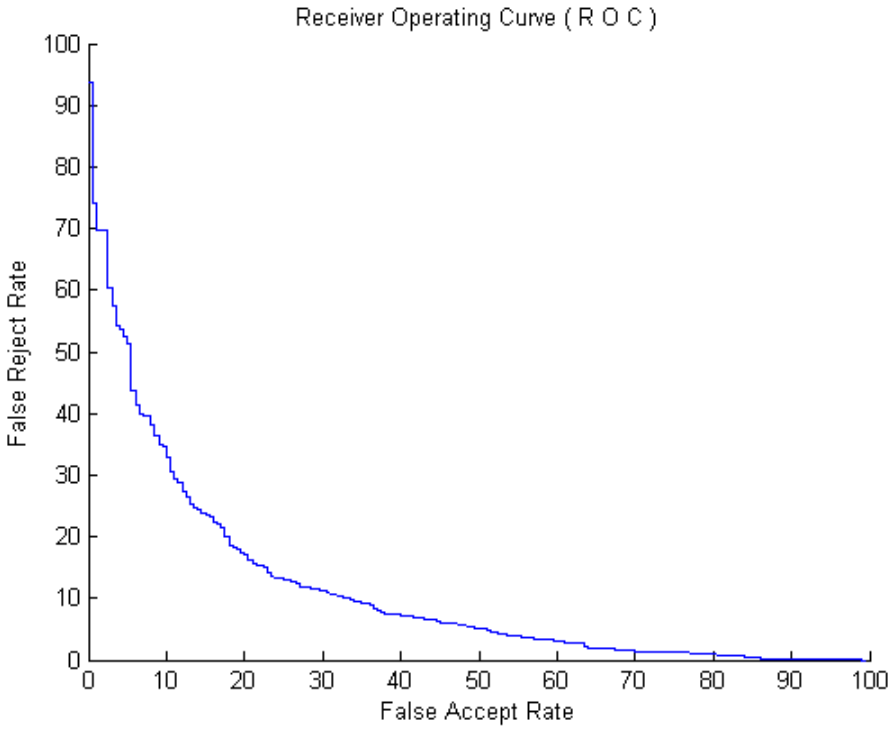

Figure 5.72: ROC Curve of 550 Zernike features tested on DB2

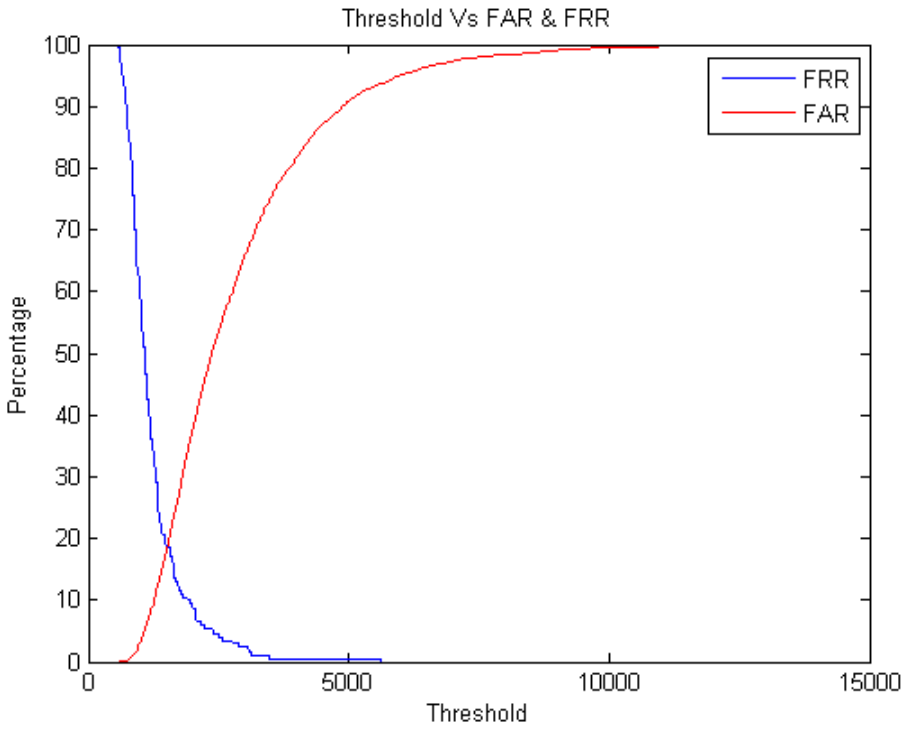

Figure 5.73: EER of 580 Zernike features tested on DB2

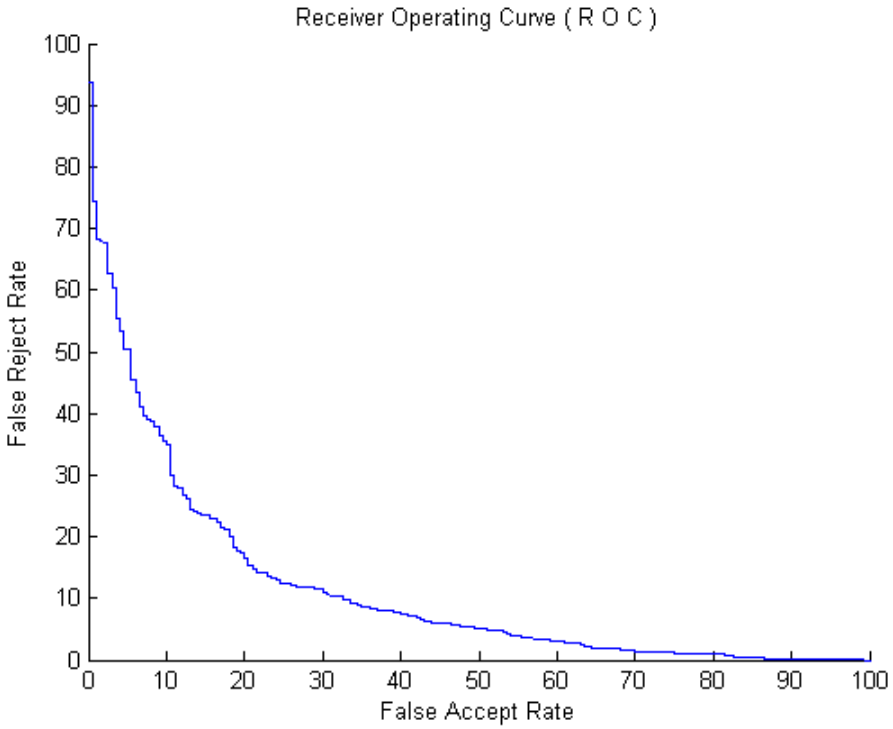

Figure 5.74: ROC Curve of 580 Zernike features tested on DB2

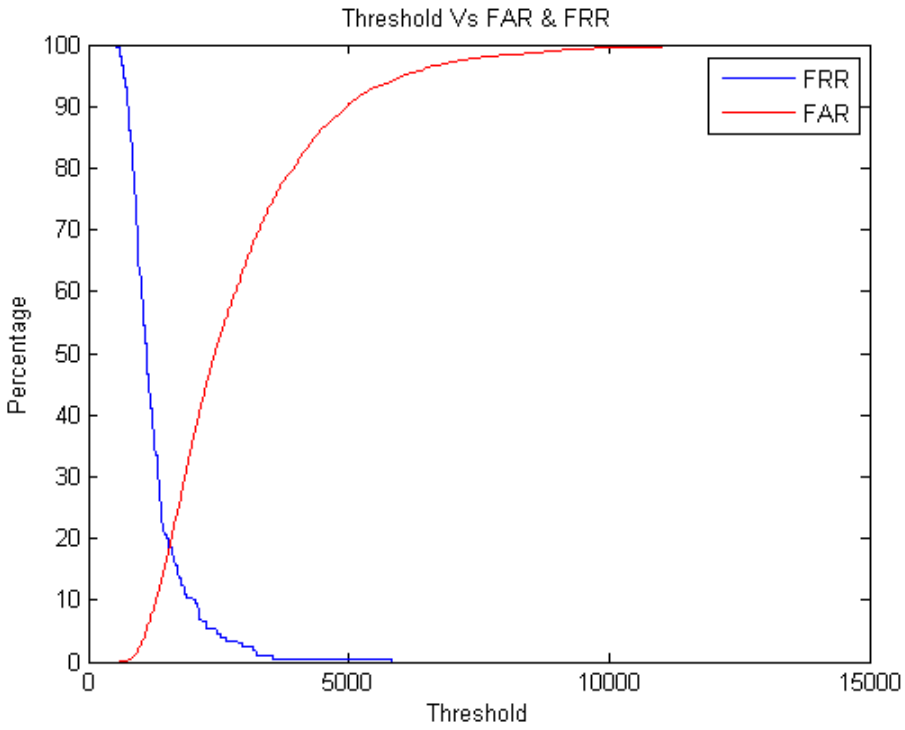

Figure 5.75: EER of 600 Zernike features tested on DB2

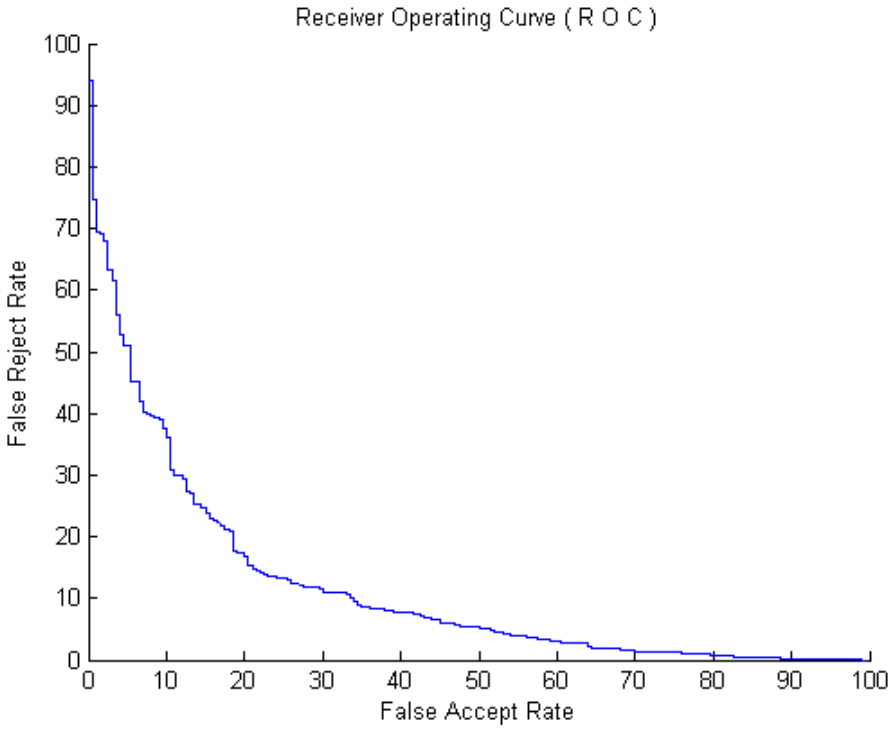

Figure 5.76: ROC Curve of 600 Zernike features tested on DB2

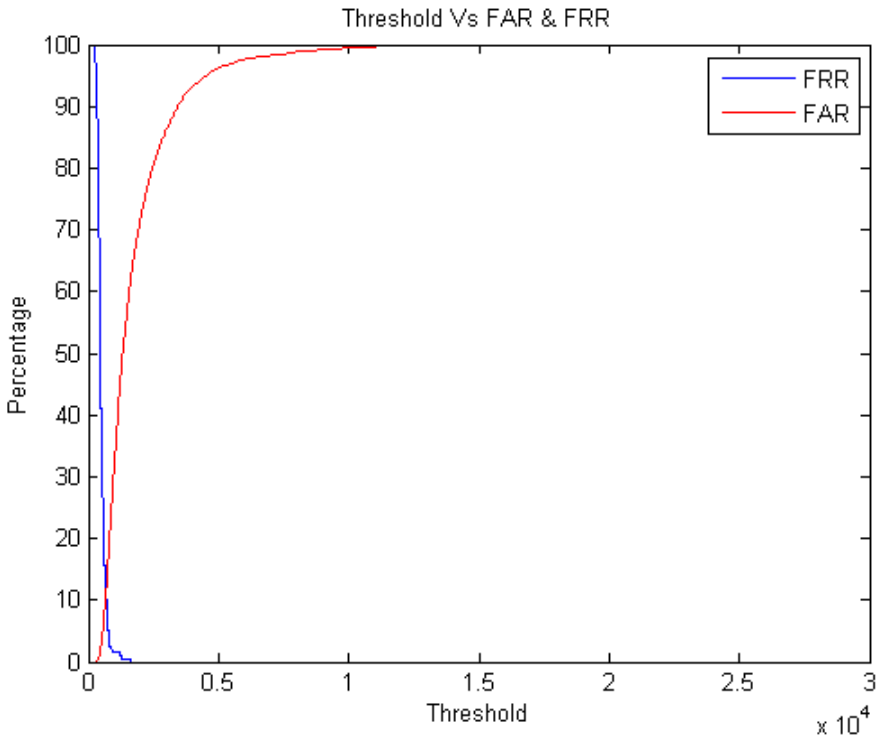

Figure 5.77: EER of 245 Zernike features tested on DB3

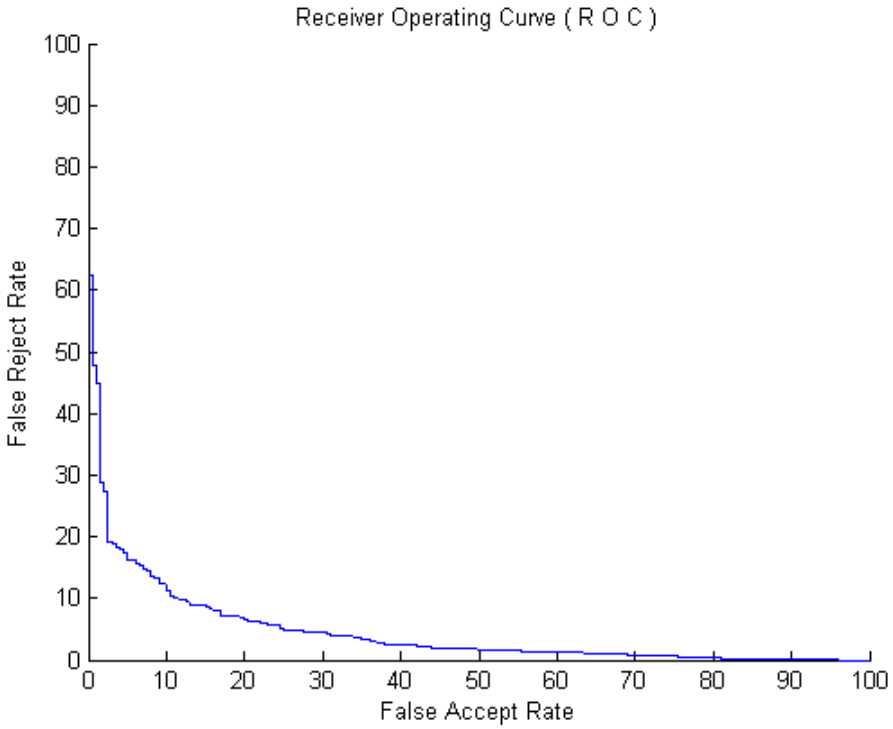

Figure 5.78: ROC Curve of 245 Zernike features tested on DB3

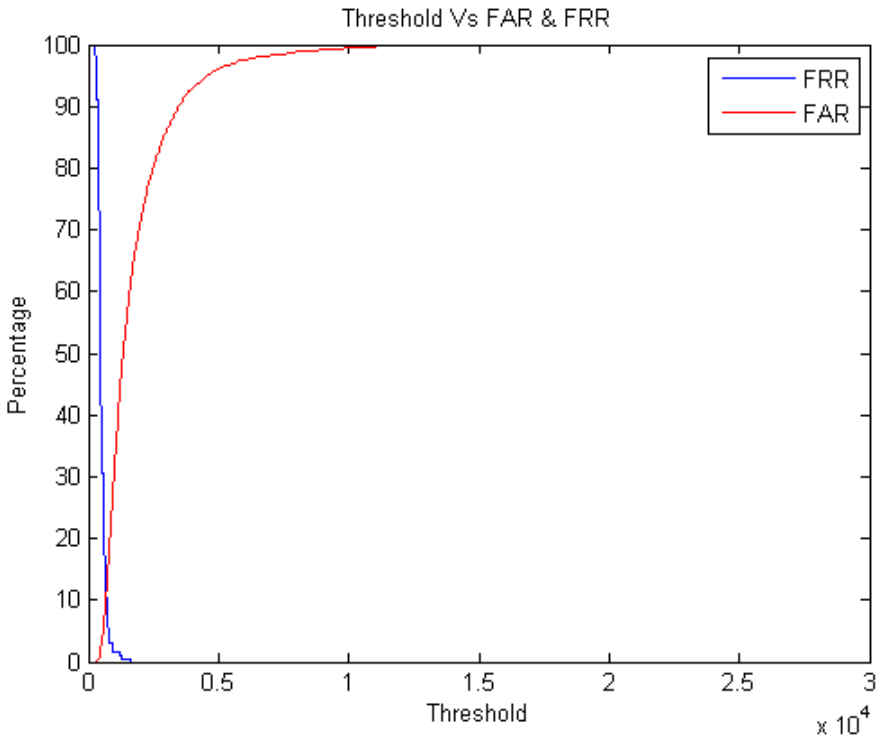

Figure 5.79: EER of 250 Zernike features tested on DB3

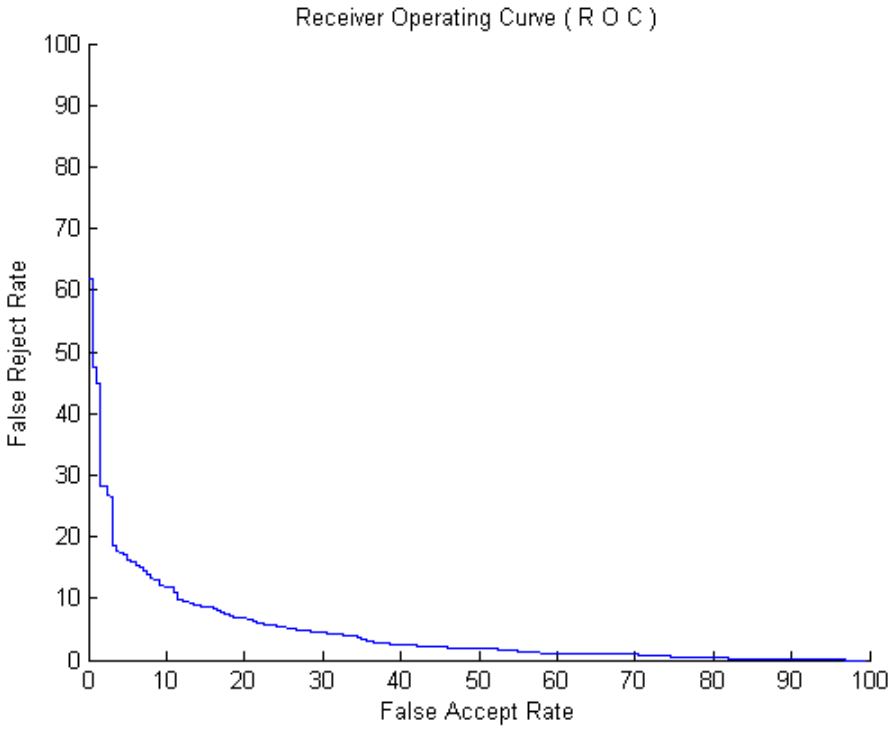

Figure 5.80: ROC Curve of 250 Zernike features tested on DB3

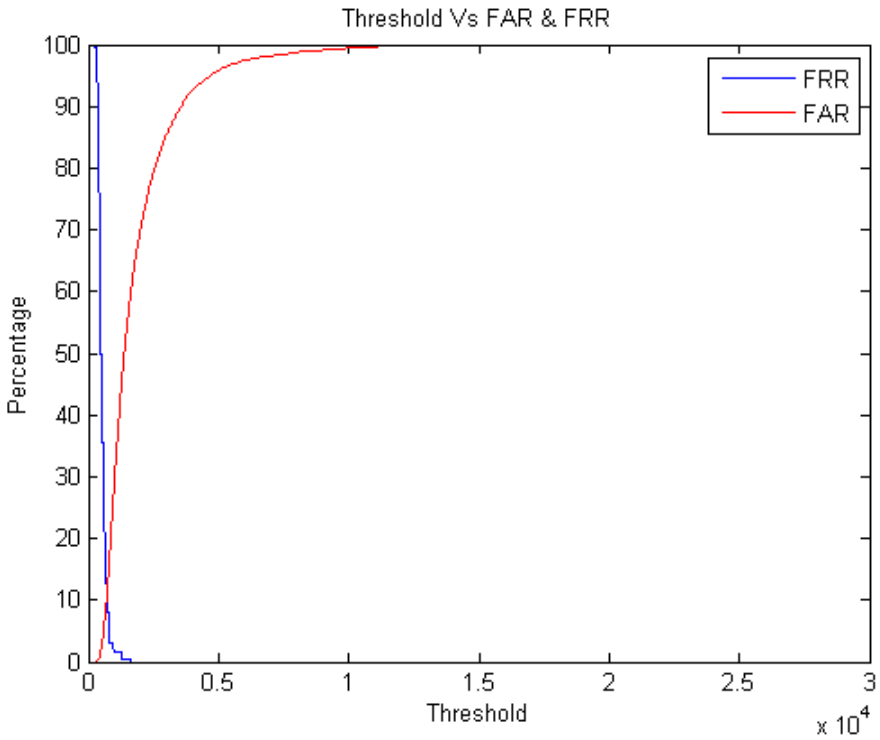

Figure 5.81: EER of 260 Zernike features tested on DB3

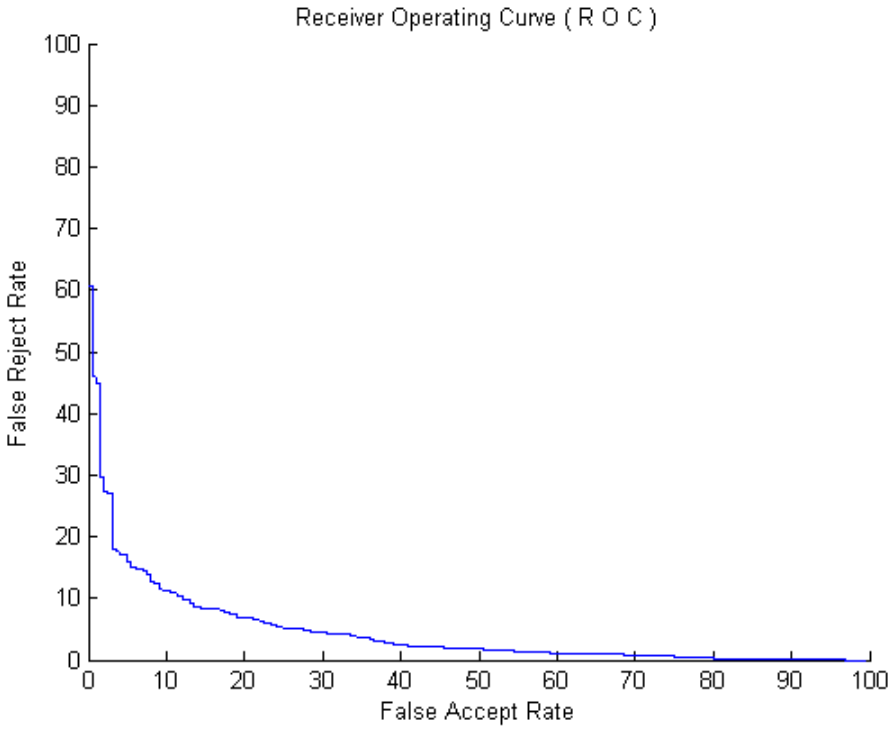

Figure 5.82: ROC Curve of 260 Zernike features tested on DB3

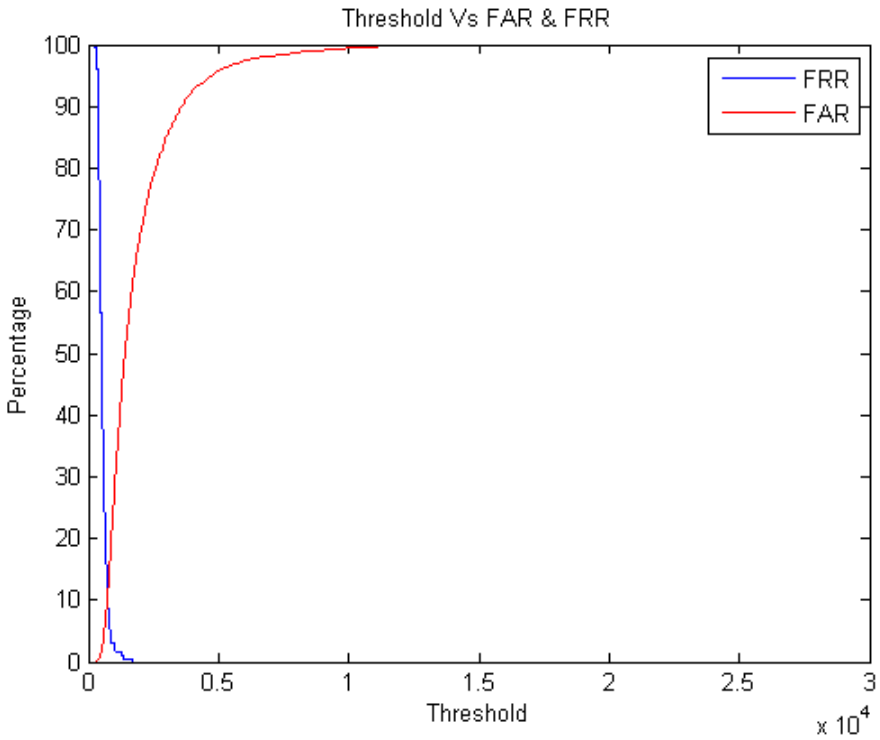

Figure 5.83: EER of 270 Zernike features tested on DB3

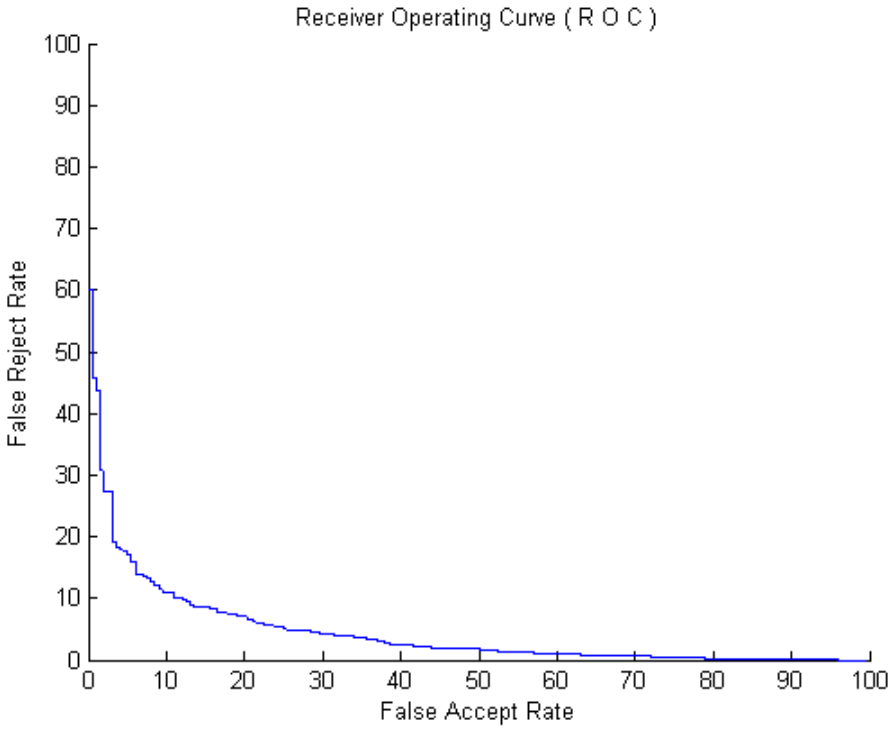

Figure 5.84: ROC Curve of 270 Zernike features tested on DB3

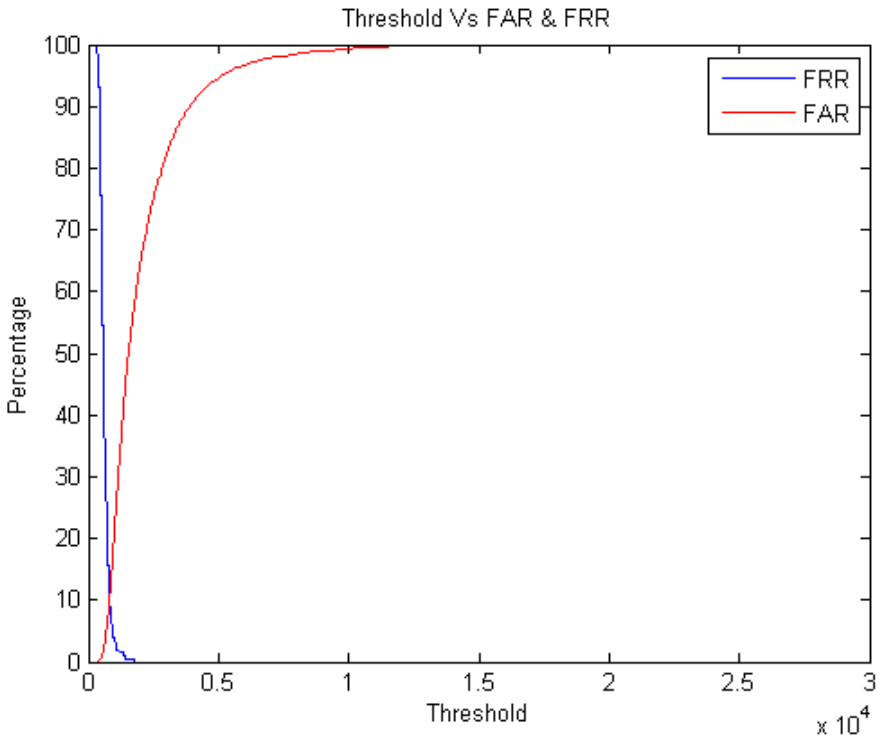

Figure 5.85: EER of 300 Zernike features tested on DB3

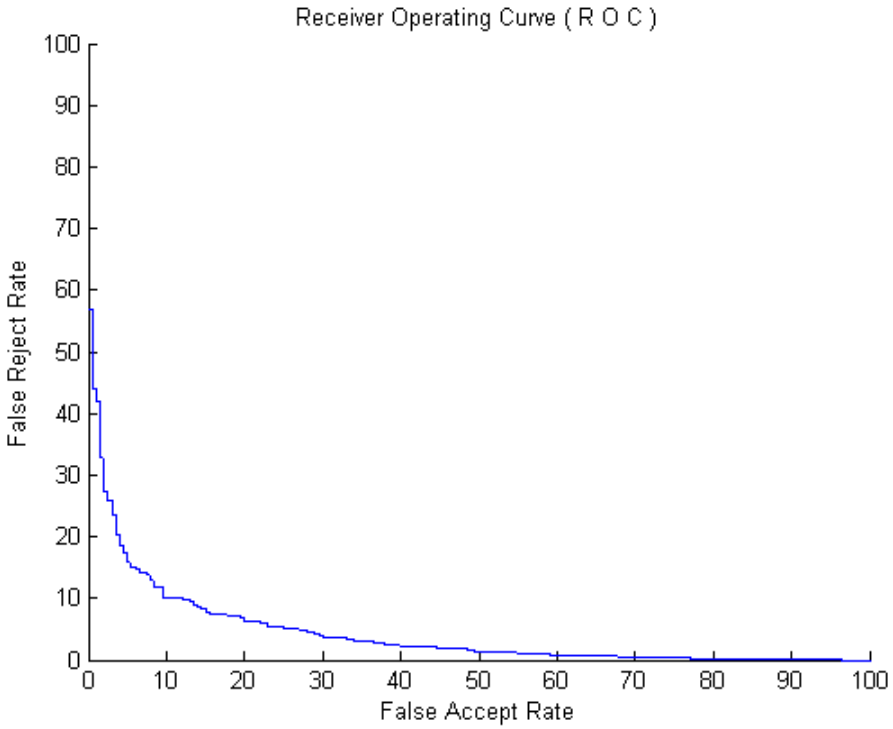

Figure 5.86: ROC Curve of 300 Zernike features tested on DB3

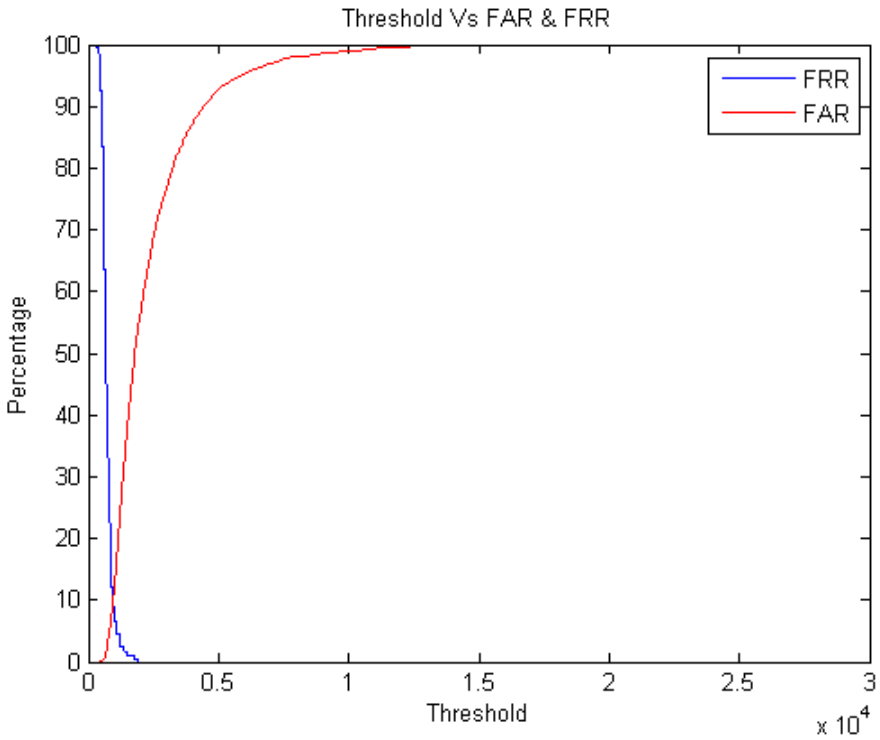

Figure 5.87: EER of 350 Zernike features tested on DB3

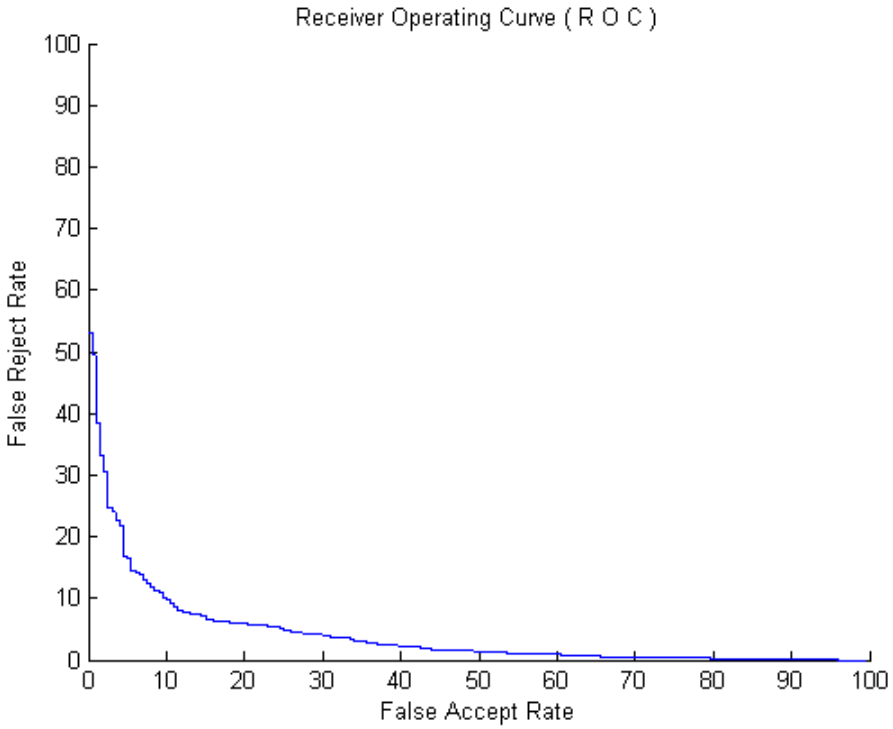

Figure 5.88: ROC Curve of 350 Zernike features tested on DB3

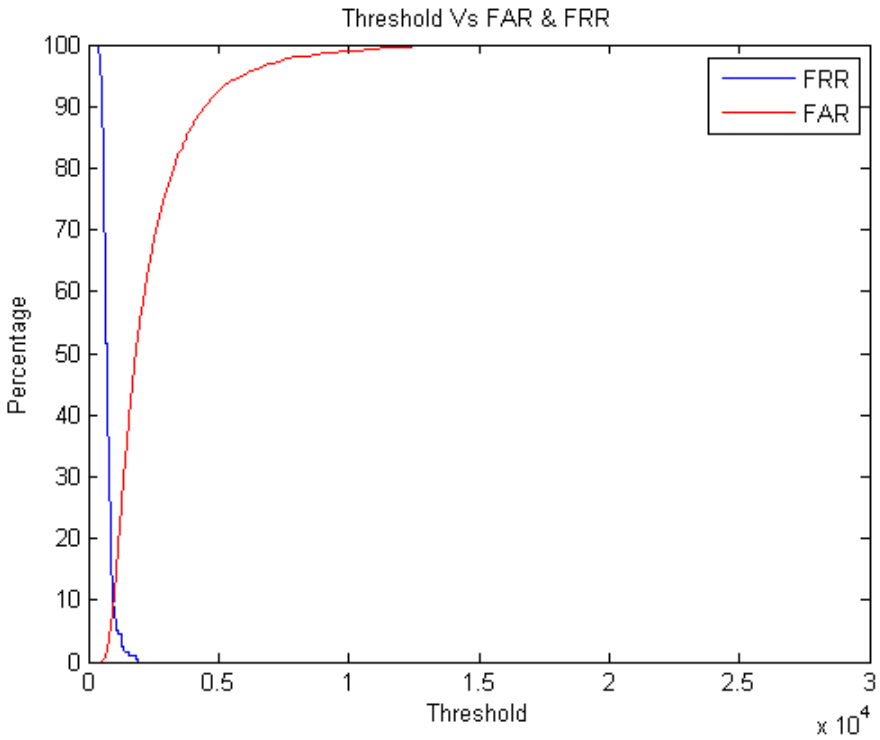

Figure 5.89: EER of 360 Zernike features tested on DB3

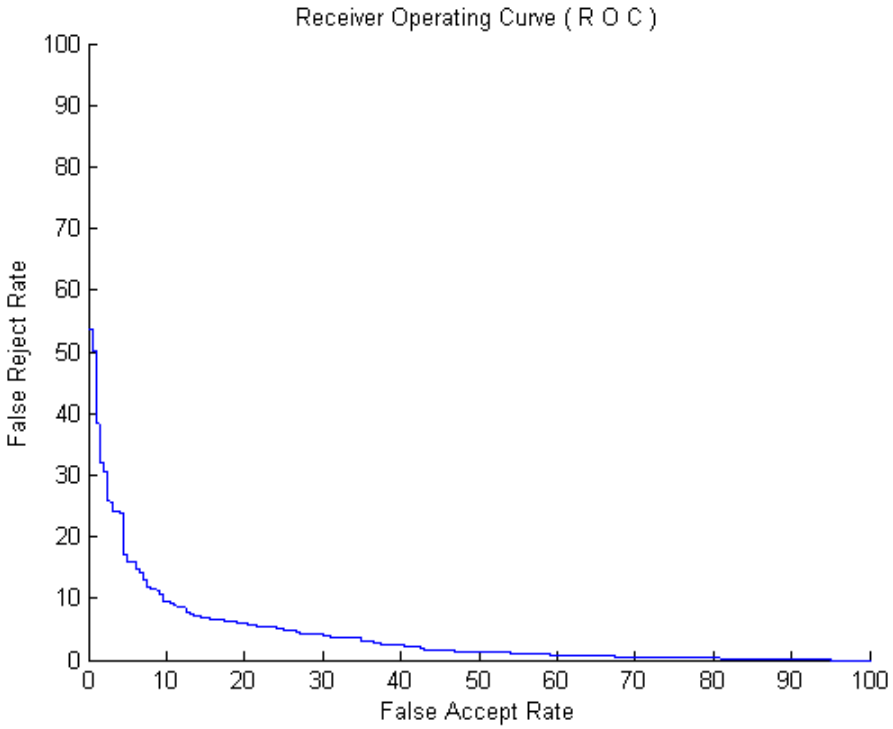

Figure 5.90: ROC Curve of 360 Zernike features tested on DB3

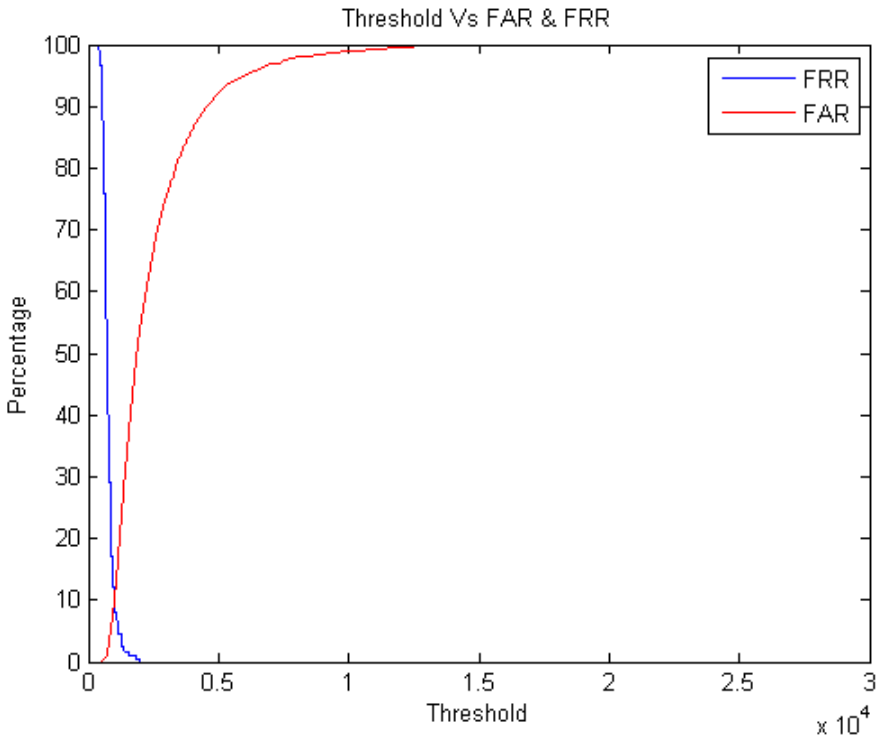

Figure 5.91: EER of 370 Zernike features tested on DB3

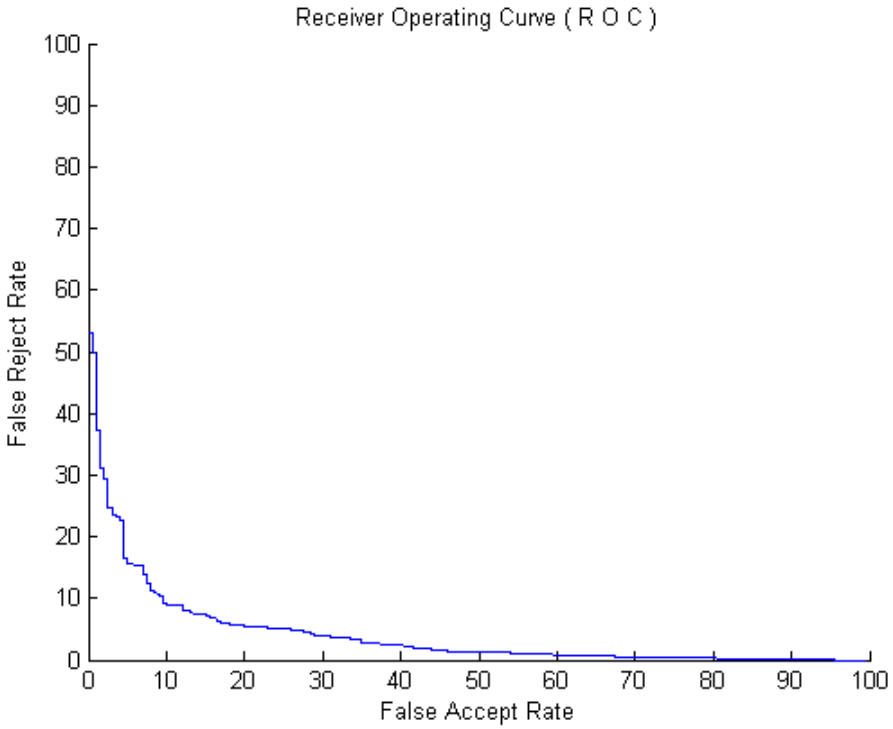

Figure 5.92: ROC Curve of 370 Zernike features tested on DB3

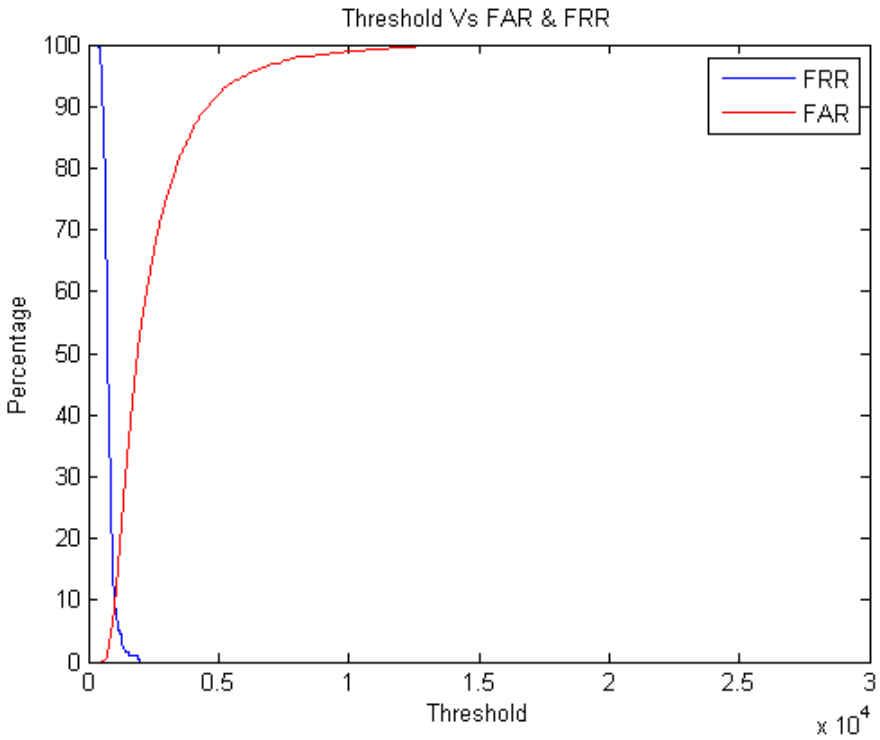

Figure 5.93: EER of 380 Zernike features tested on DB3

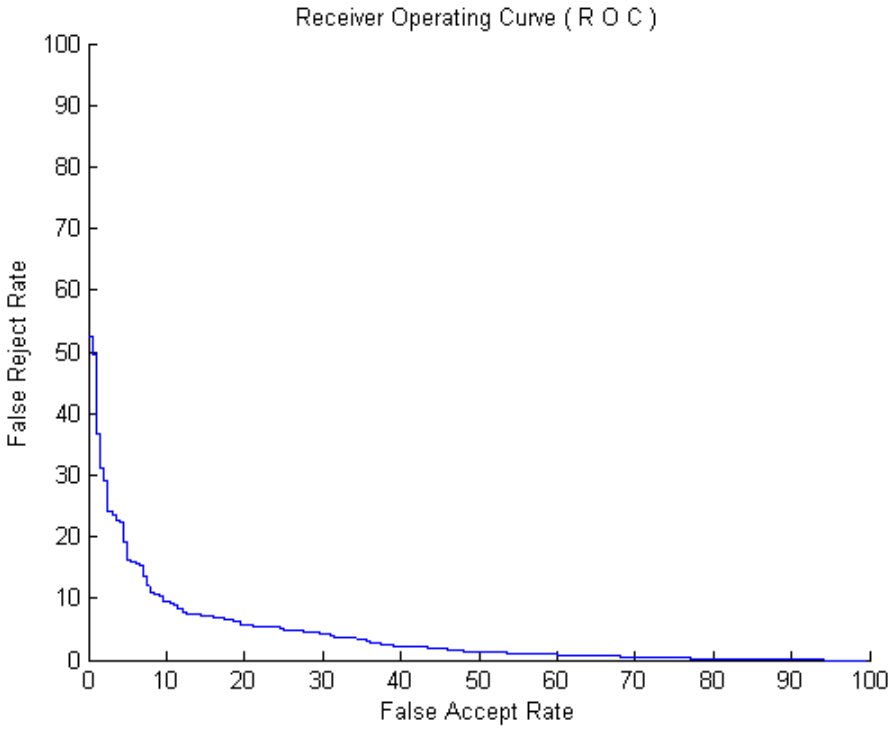

Figure 5.94: ROC Curve of 380 Zernike features tested on DB3

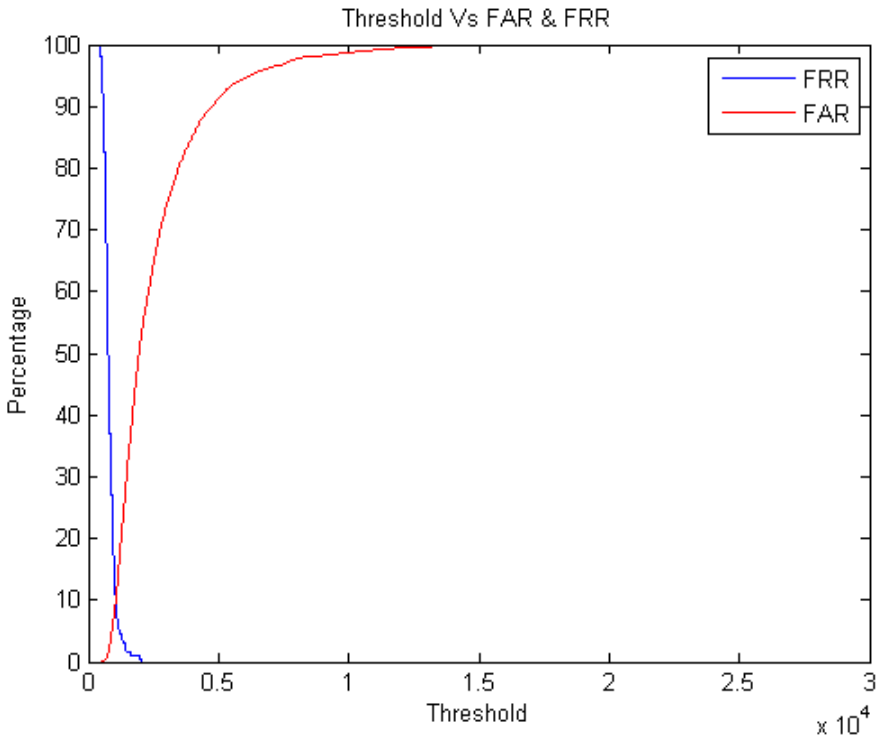

Figure 5.95: EER of 390 Zernike features tested on DB3

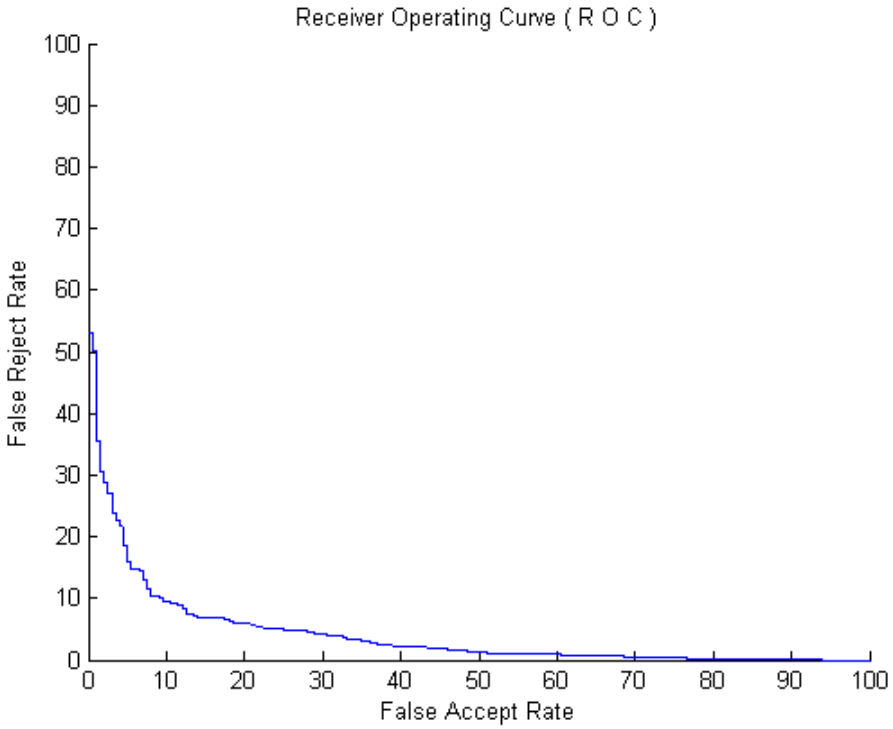

Figure 5.96: ROC Curve of 390 Zernike features tested on DB3

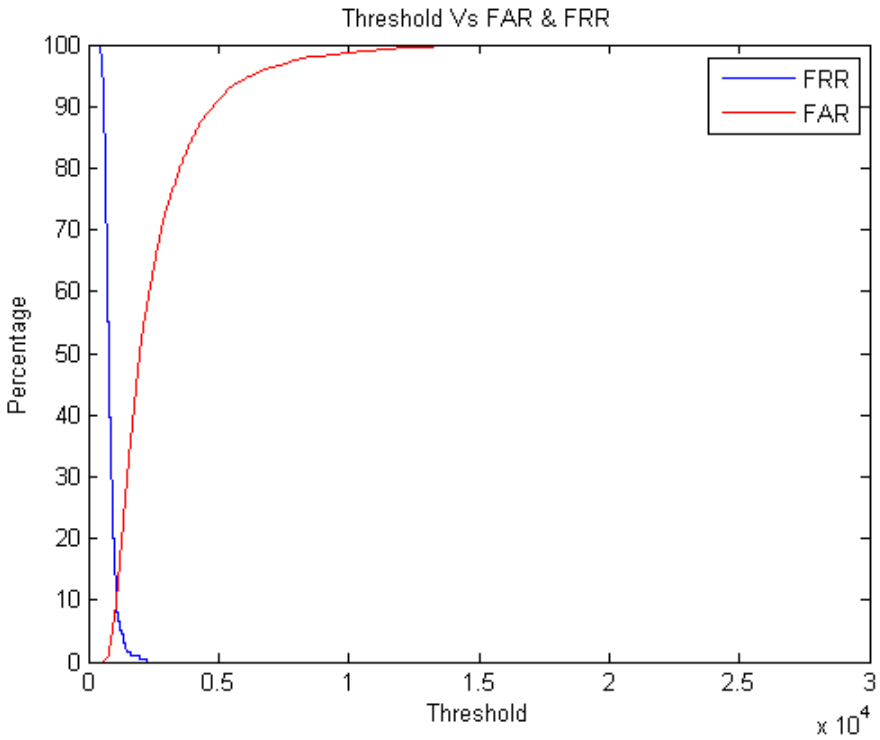

Figure 5.97: EER of 400 Zernike features tested on DB3

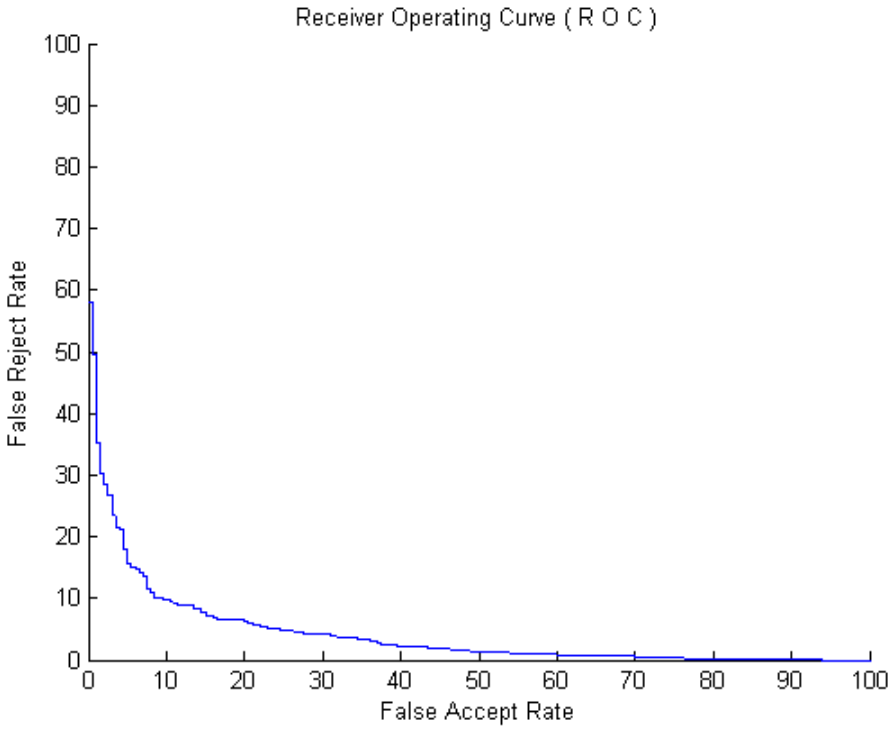

Figure 5.98: ROC Curve of 400 Zernike features tested on DB3

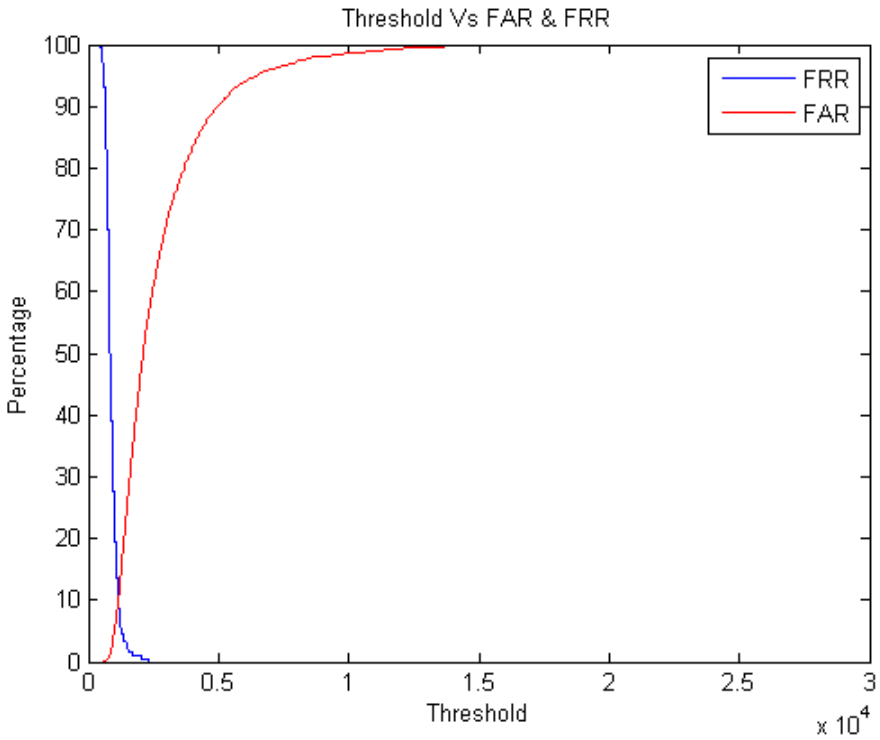

Figure 5.99: EER of 425 Zernike features tested on DB3

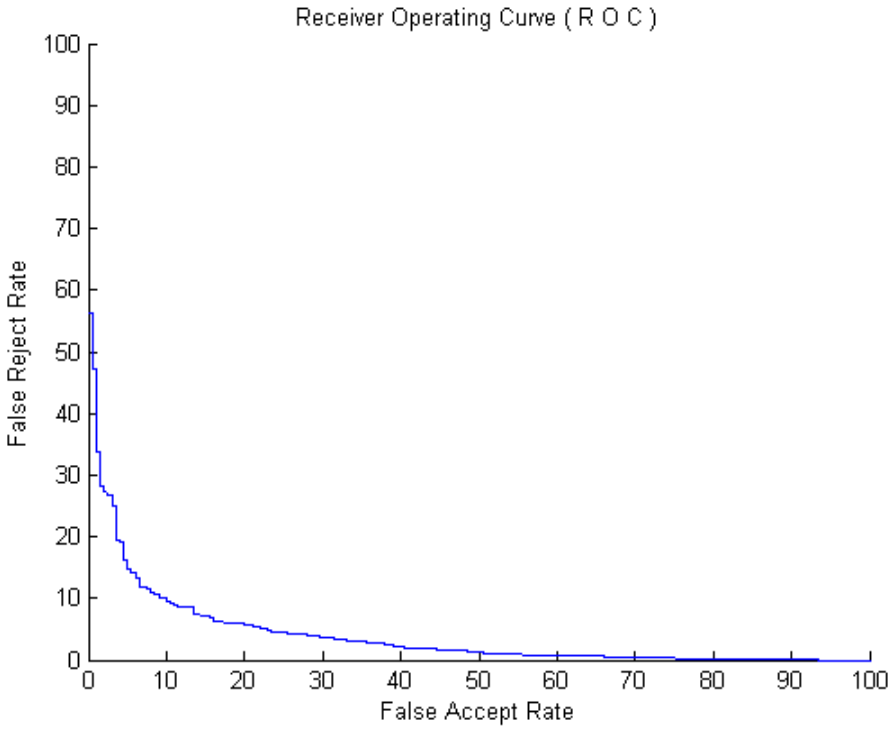

Figure 5.100: ROC Curve of 425 Zernike features tested on DB3

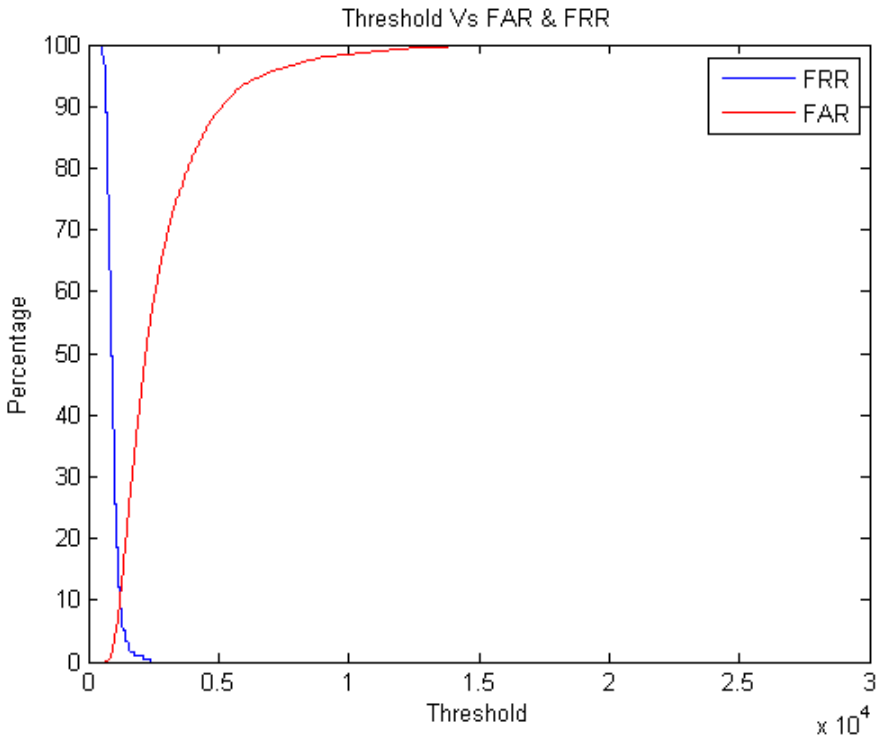

Figure 5.101: EER of 450 Zernike features tested on DB3

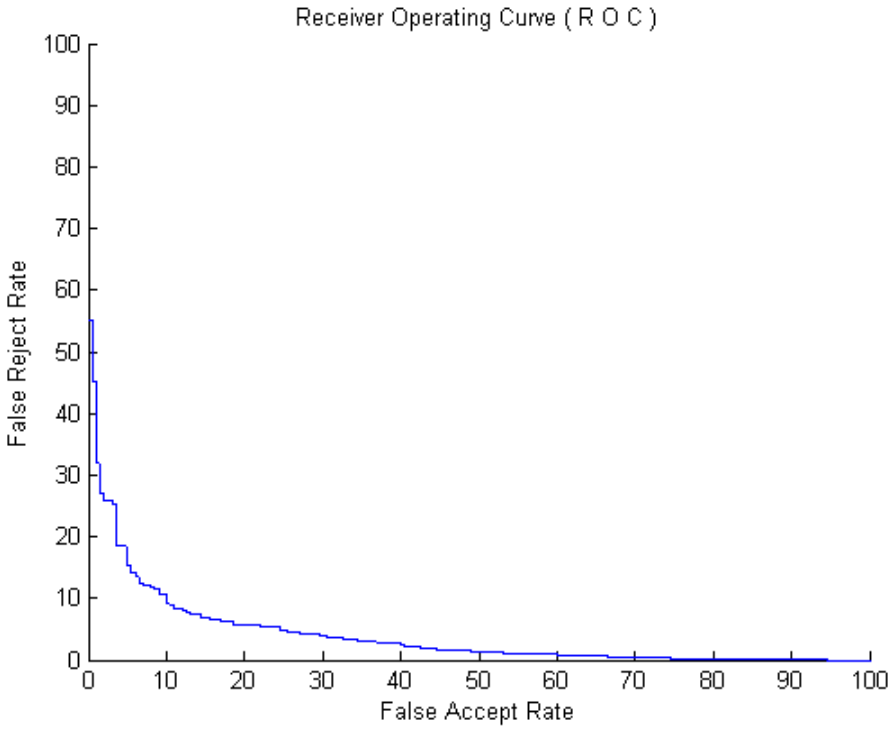

Figure 5.102: ROC Curve of 450 Zernike features tested on DB3

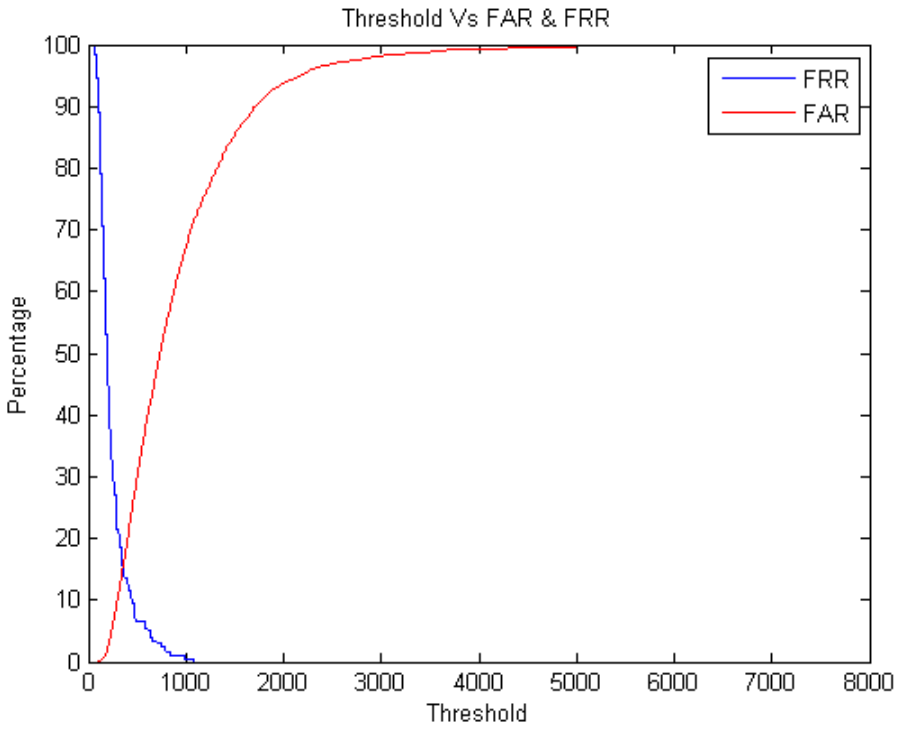

Figure 5.103: EER of 100 Zernike features tested on DB4

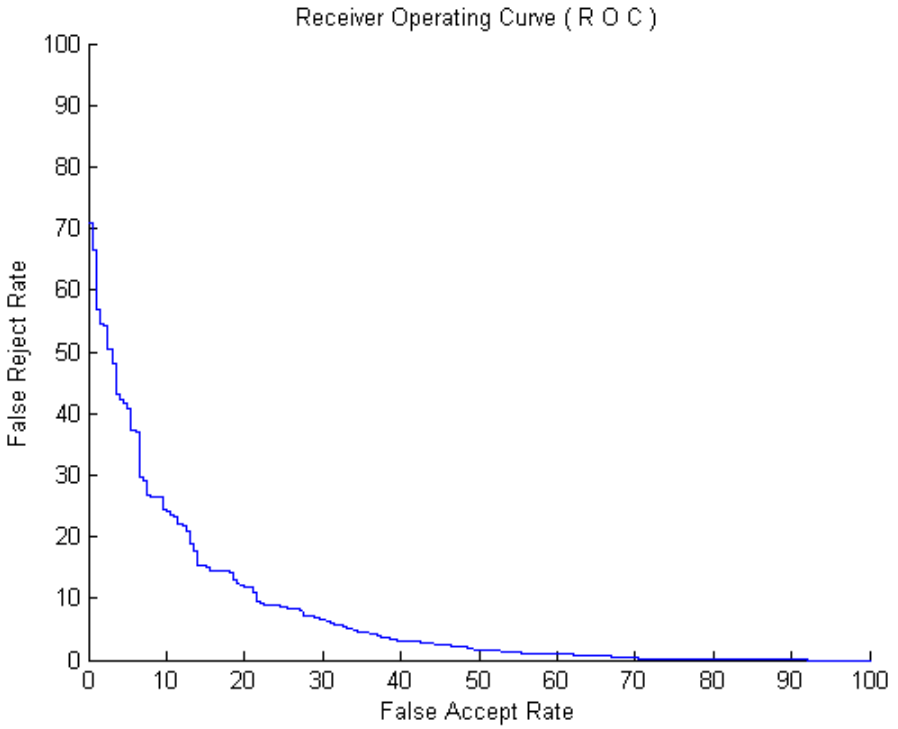

Figure 5.104: ROC Curve of 100 Zernike features tested on DB4

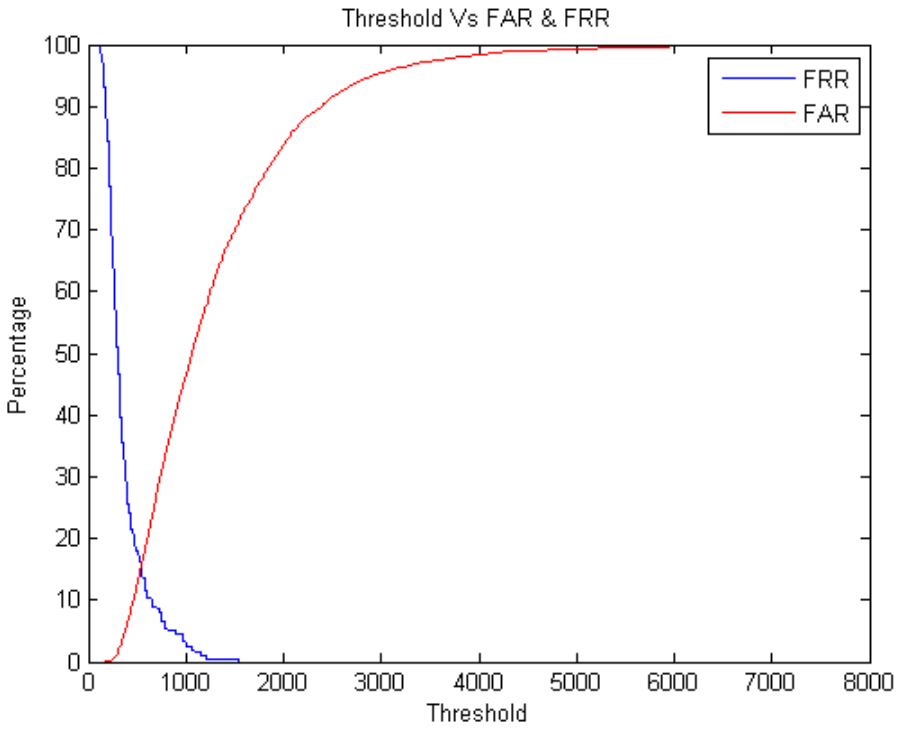

Figure 5.105: EER of 150 Zernike features tested on DB4

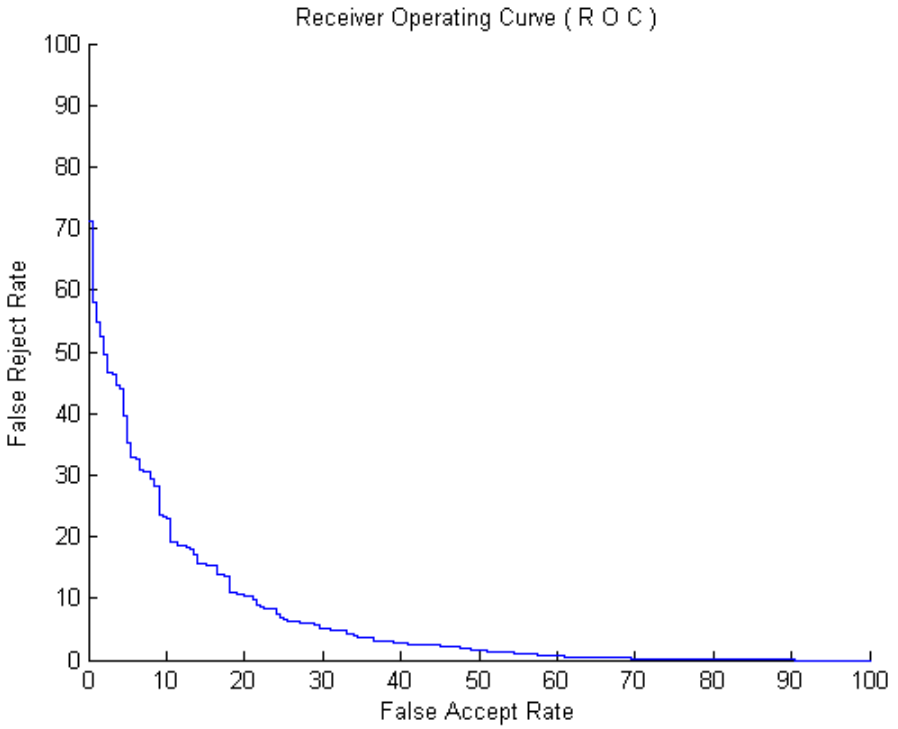

Figure 5.106: ROC Curve of 150 Zernike features tested on DB4

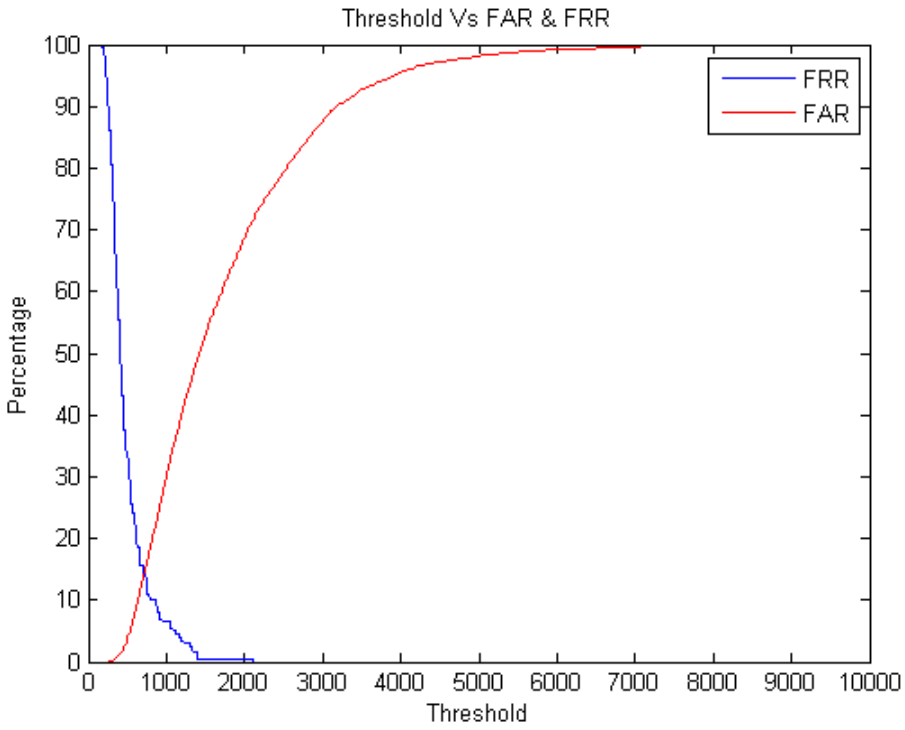

Figure 5.107: EER of 200 Zernike features tested on DB4

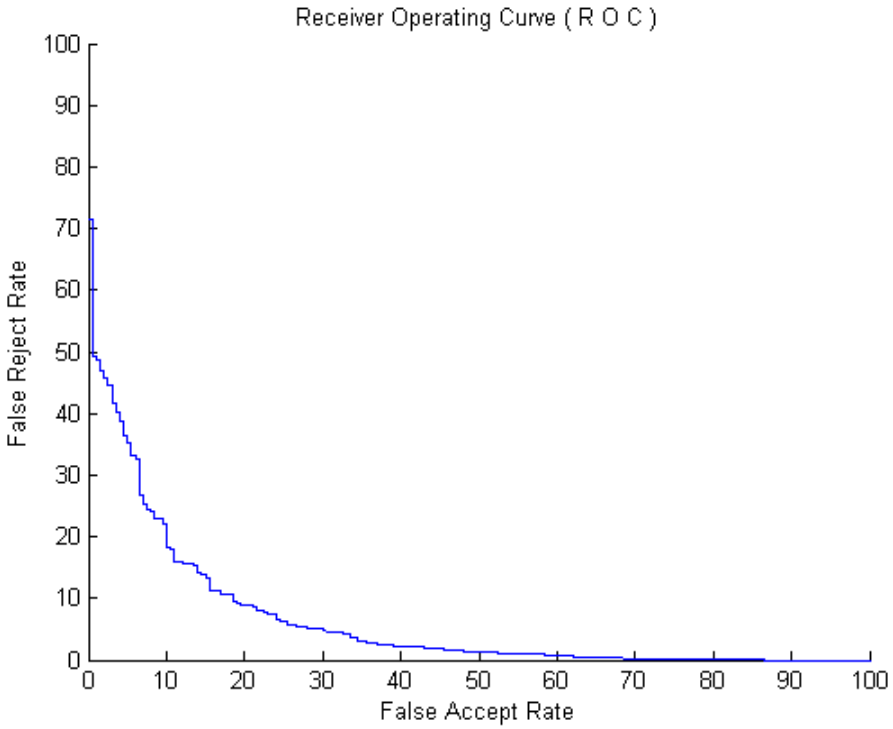

Figure 5.108: ROC Curve of 200 Zernike features tested on DB4

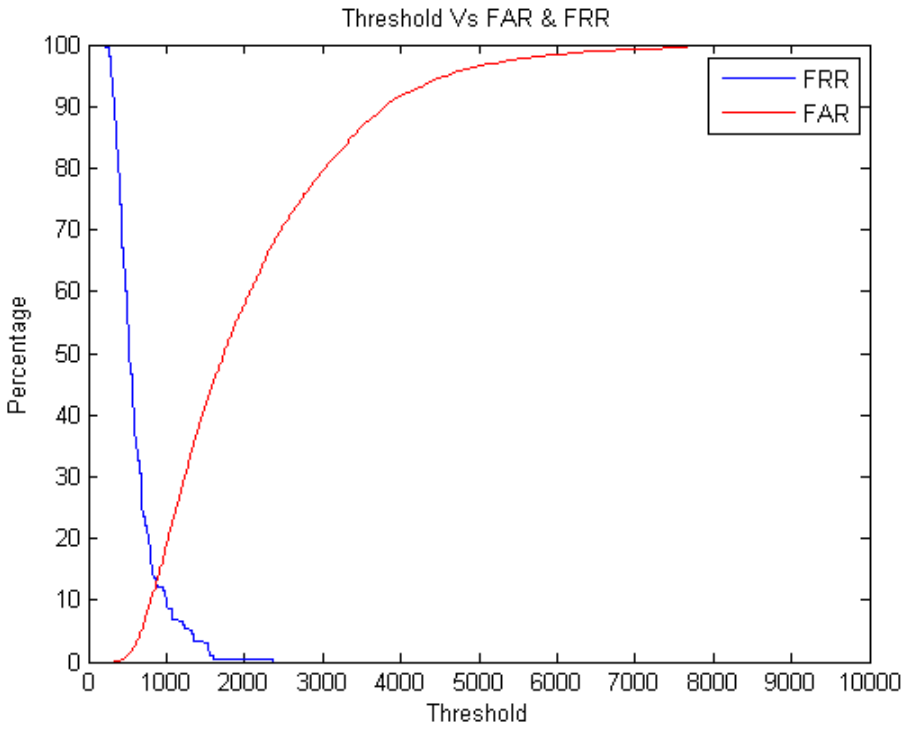

Figure 5.109: EER Curve of 250 Zernike features tested on DB4

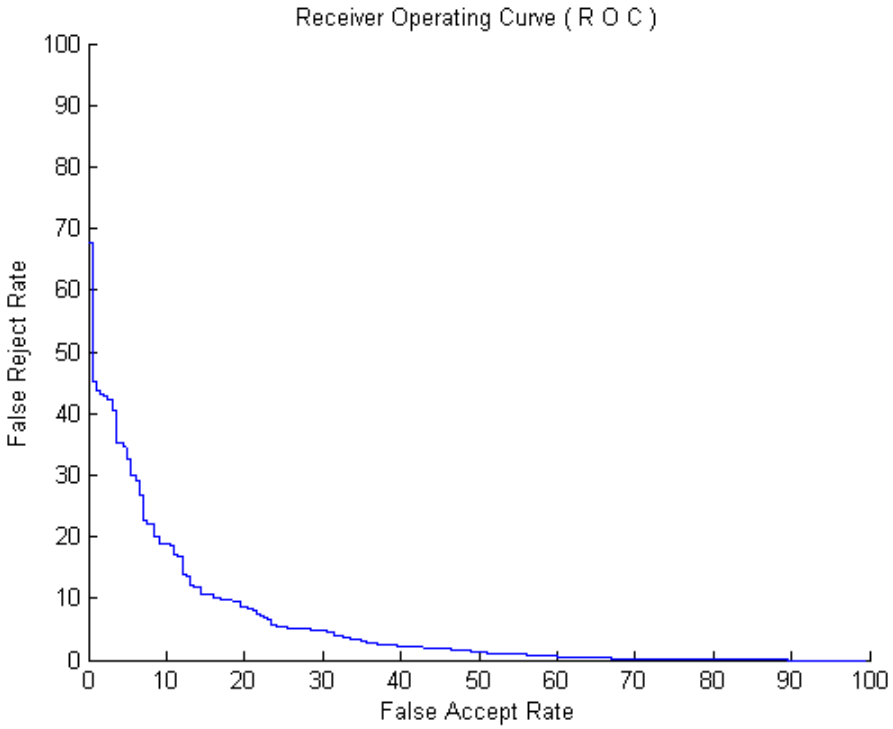

Figure 5.110: ROC Curve of 250 Zernike features tested on DB4

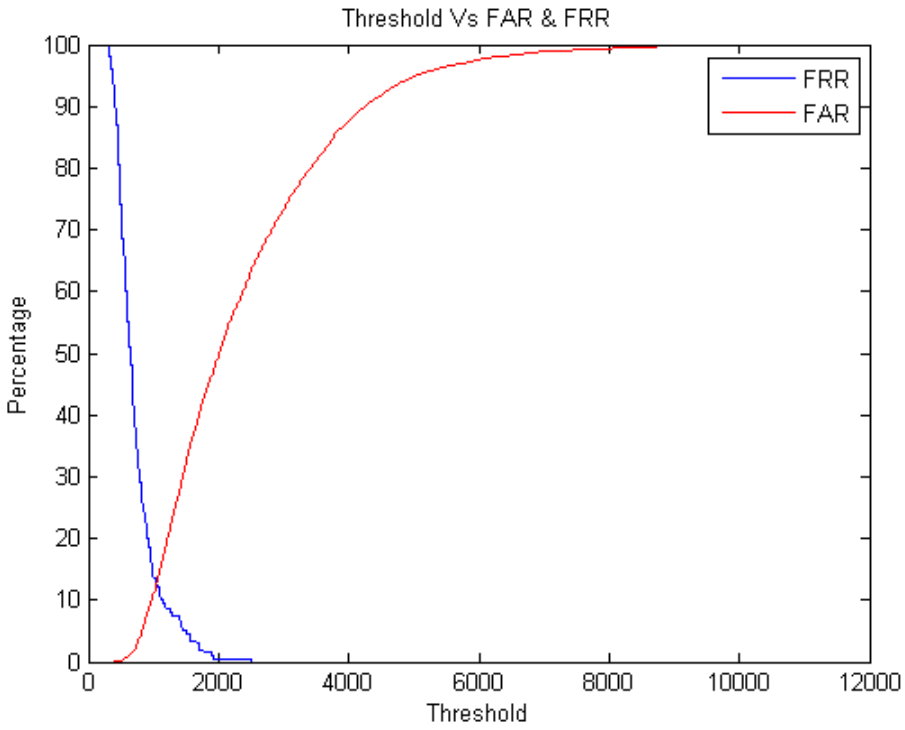

Figure 5.111: EER of 300 Zernike features tested on DB4

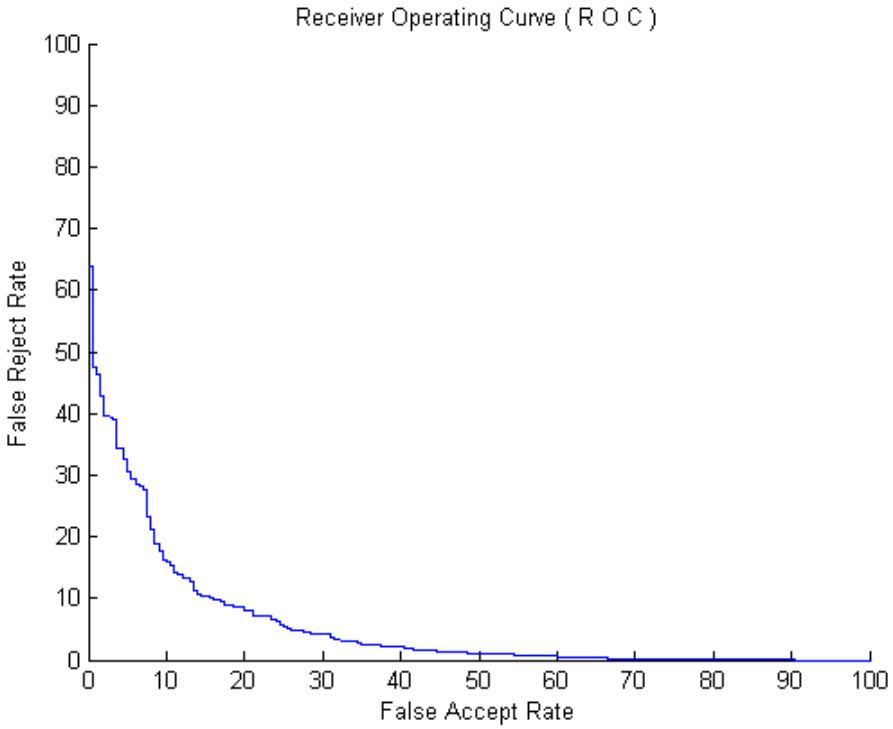

Figure 5.112: ROC Curve of 300 Zernike features tested on DB4

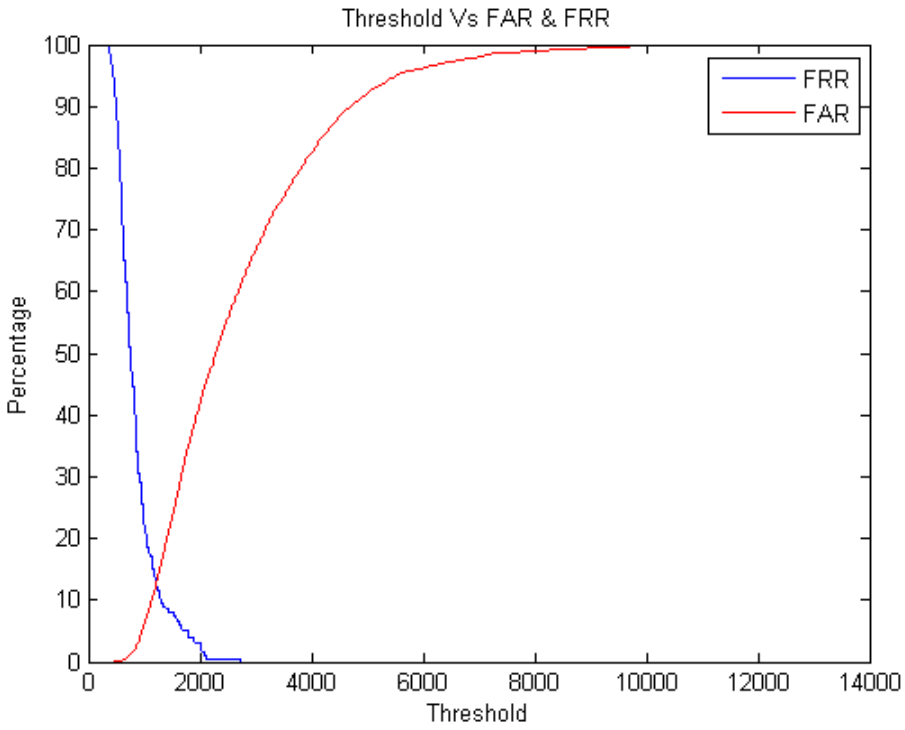

Figure 5.113: EER of 350 Zernike features tested on DB4

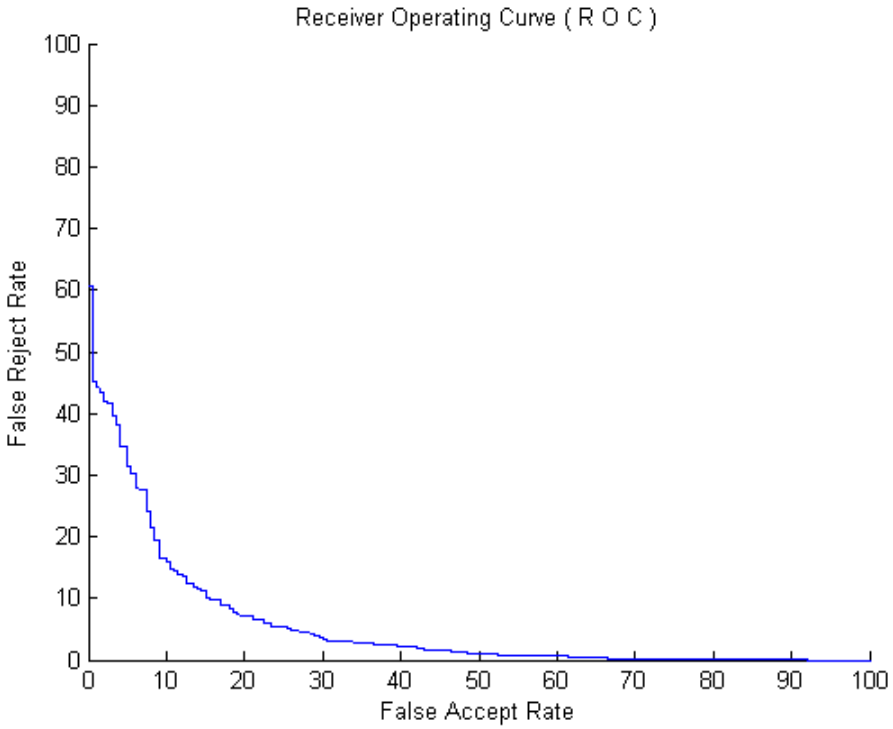

Figure 5.114: ROC Curve of 350 Zernike features tested on DB4

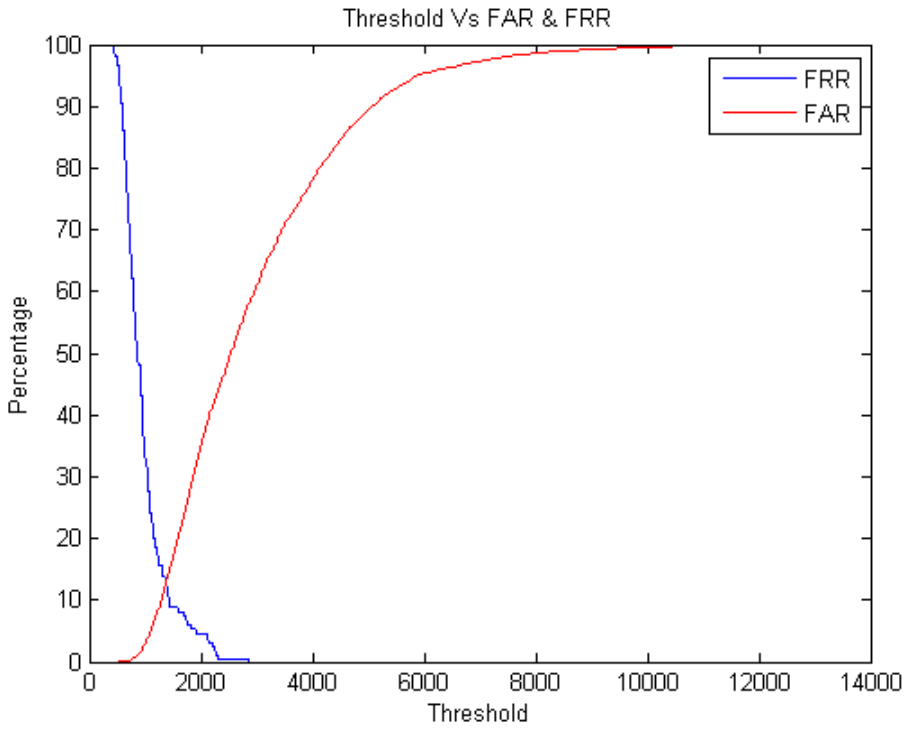

Figure 5.115: EER of 398 Zernike features tested on DB4

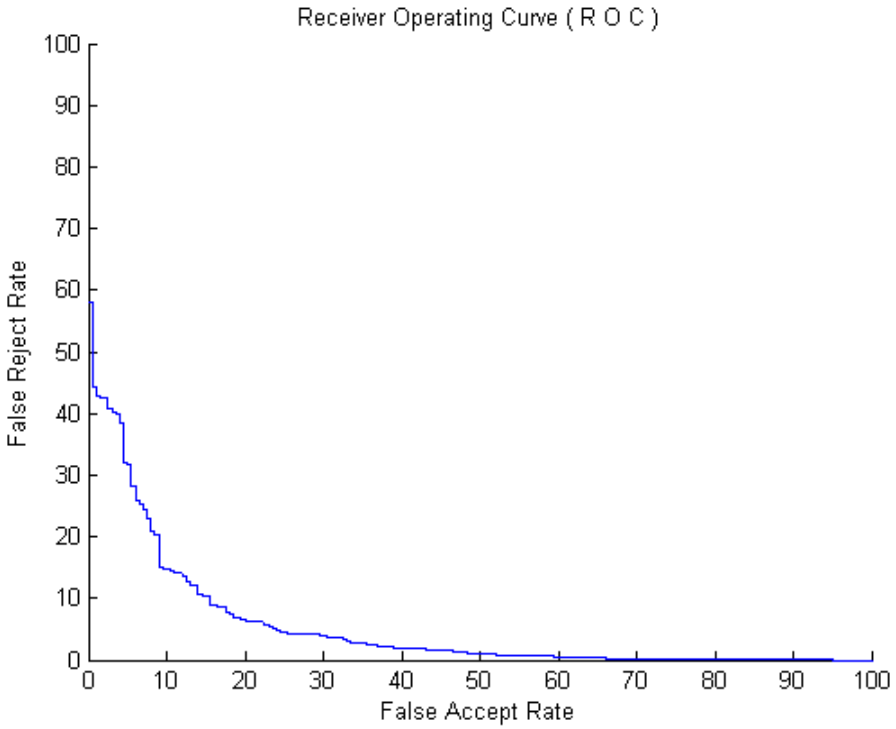

Figure 5.116: ROC Curve of 398 Zernike features tested on DB4

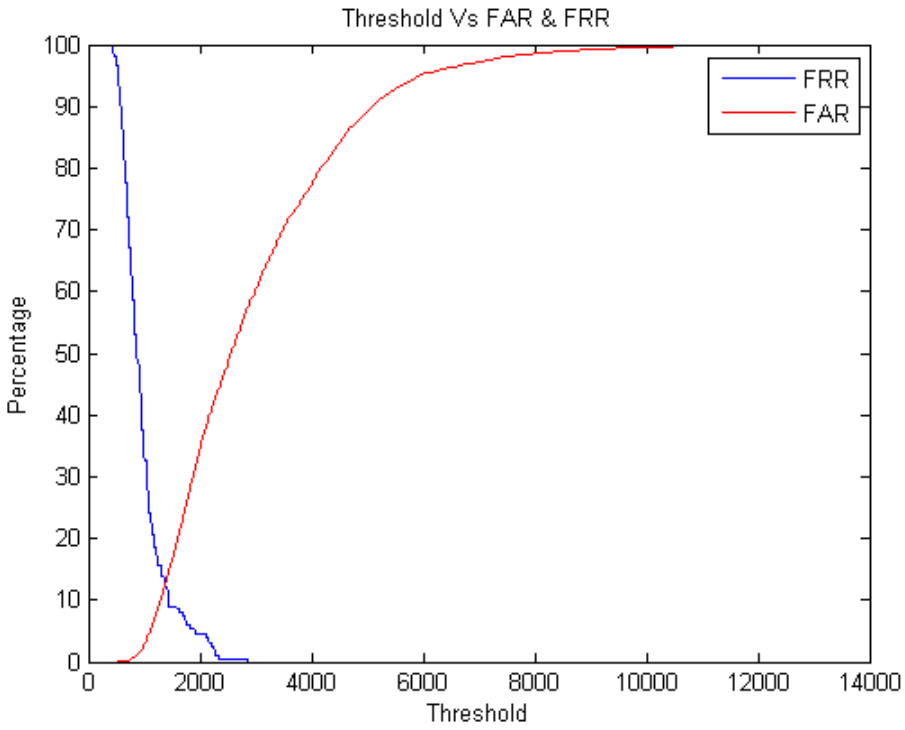

Figure 5.117: EER of 400 Zernike features tested on DB4

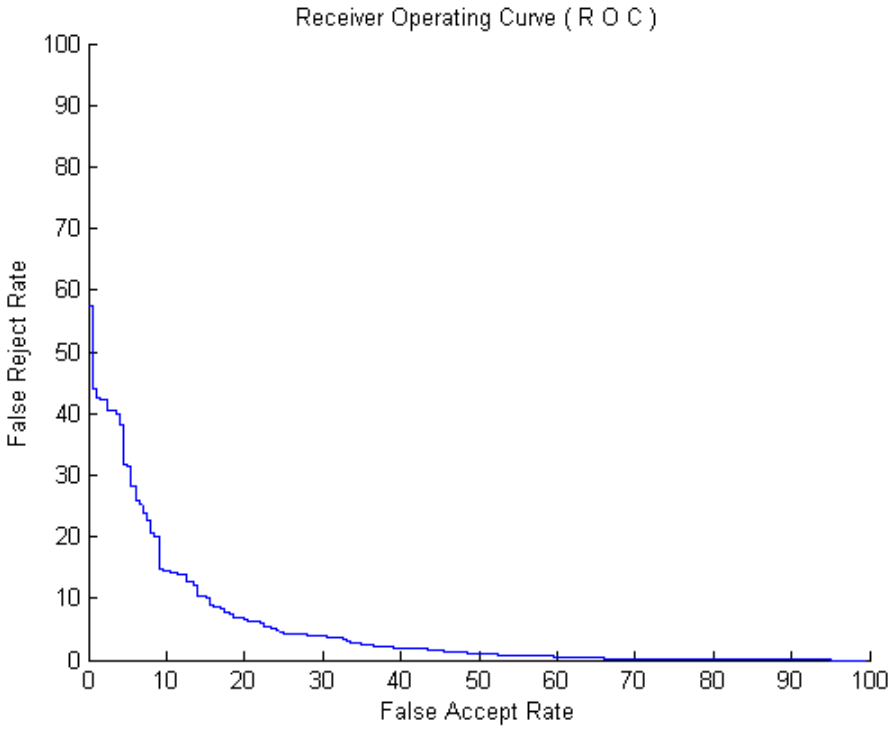

Figure 5.118: ROC Curve of 400 Zernike features tested on DB4

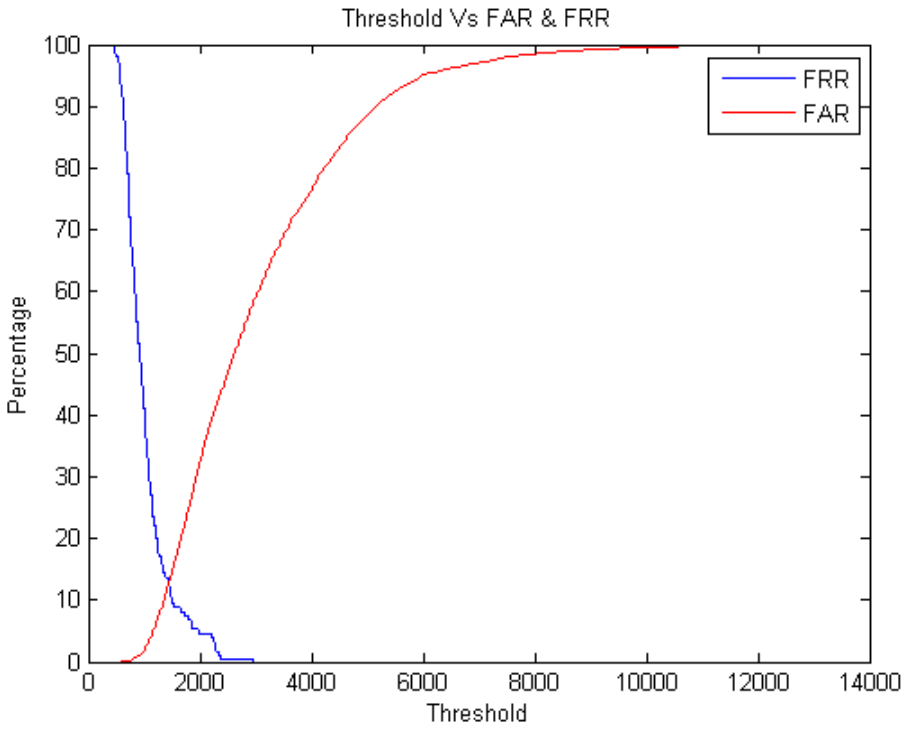

Figure 5.119: EER of 420 Zernike features tested on DB4

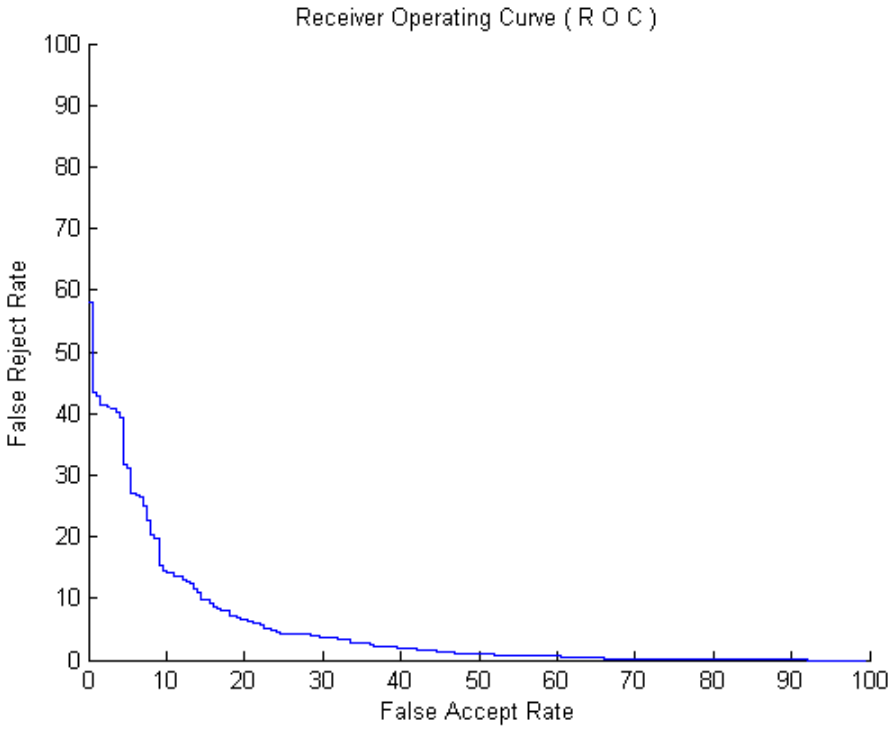

Figure 5.120: ROC Curve of 420 Zernike features tested on DB4

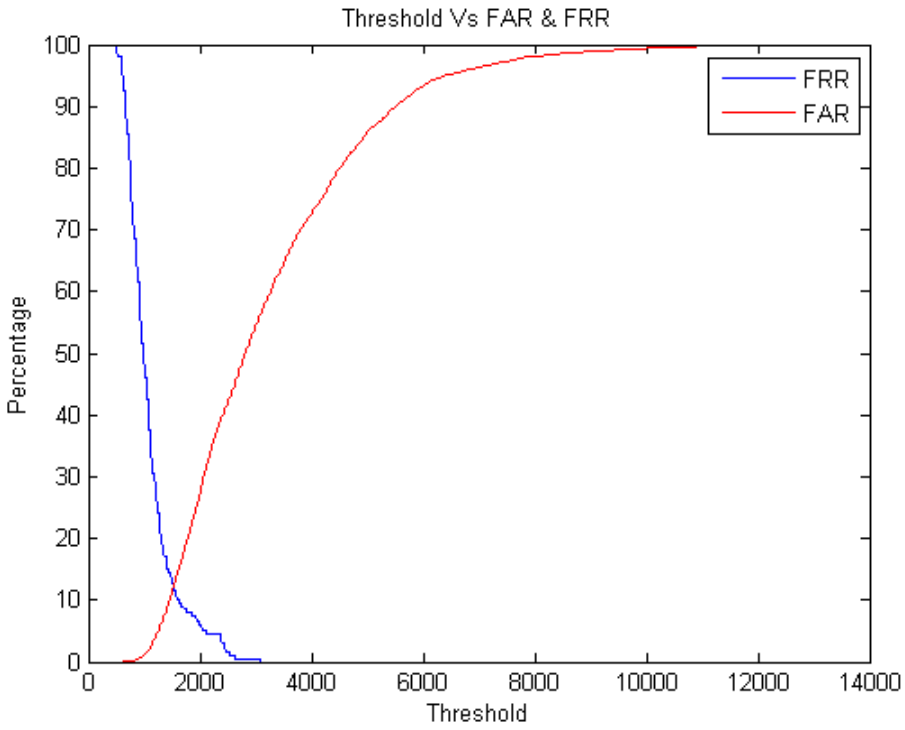

Figure 5.121: EER of 450 Zernike features tested on DB4

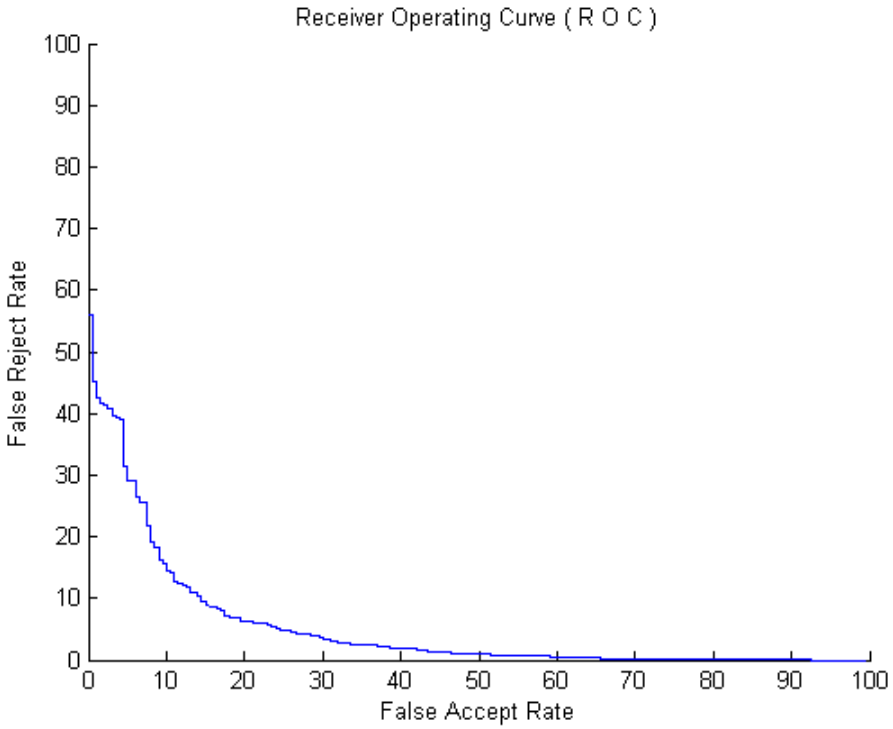

Figure 5.122: ROC Curve of 450 Zernike features tested on DB4

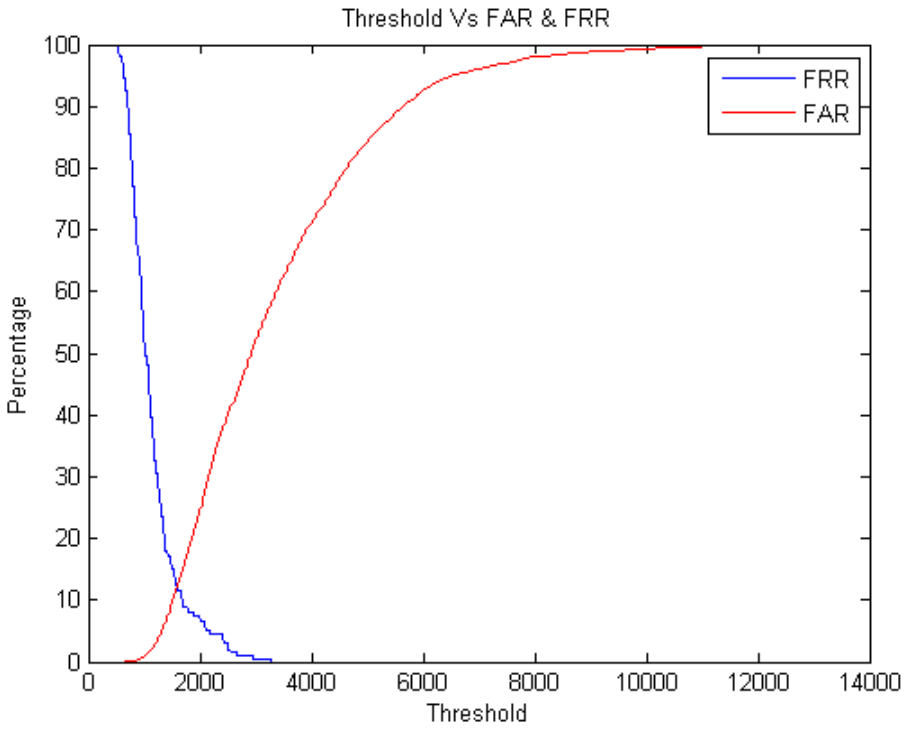

Figure 5.123: EER of 470 Zernike features tested on DB4

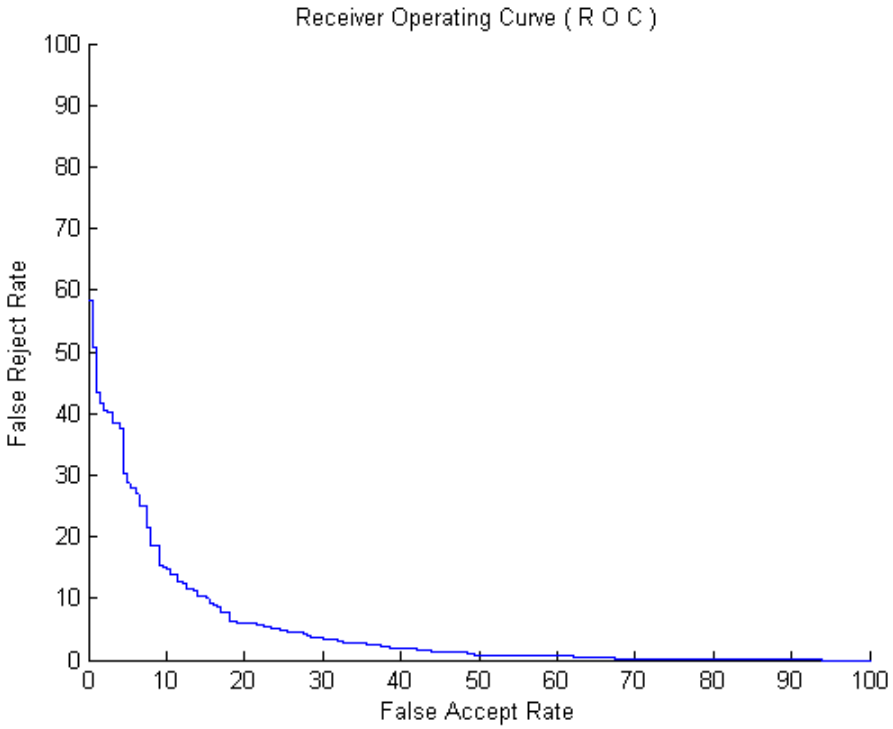

Figure 5.124: ROC Curve of 470 Zernike features tested on DB4

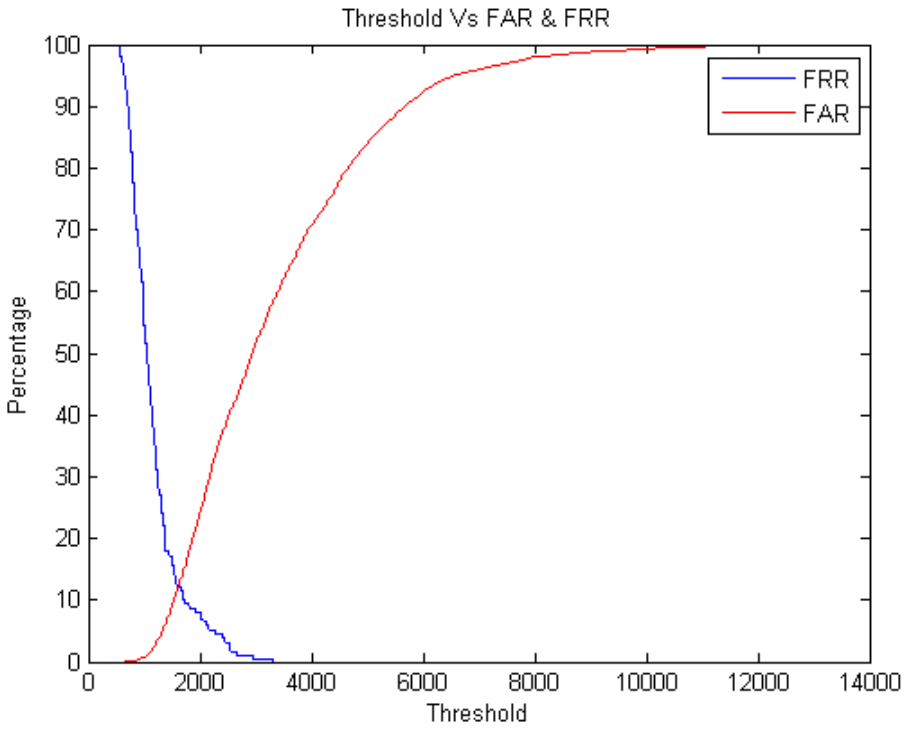

Figure 5.125: EER of 475 Zernike features tested on DB4

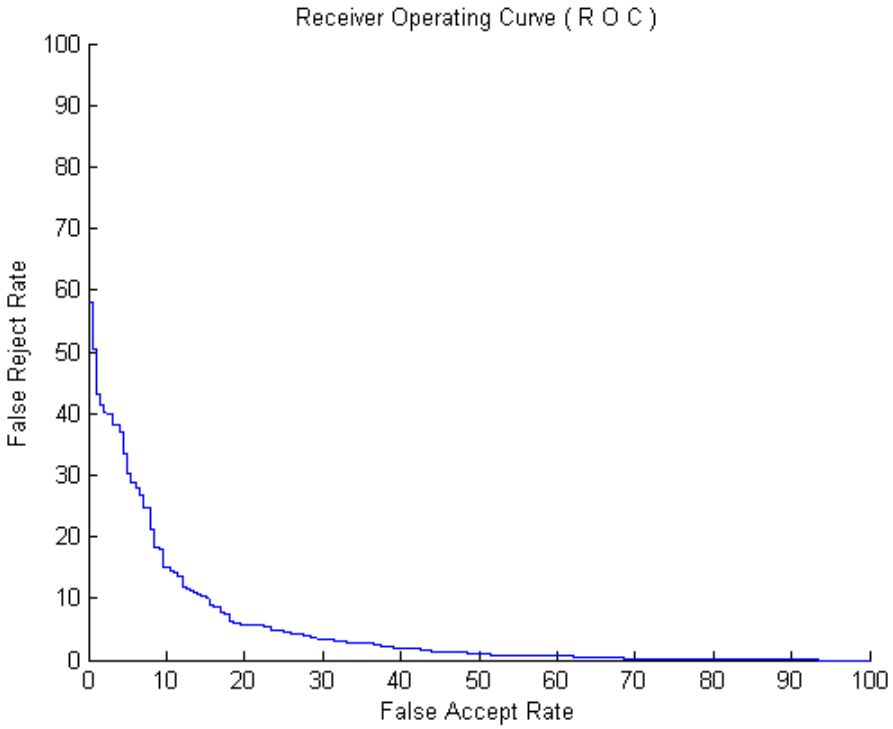

Figure 5.126: ROC Curve of 475 Zernike features tested on DB4

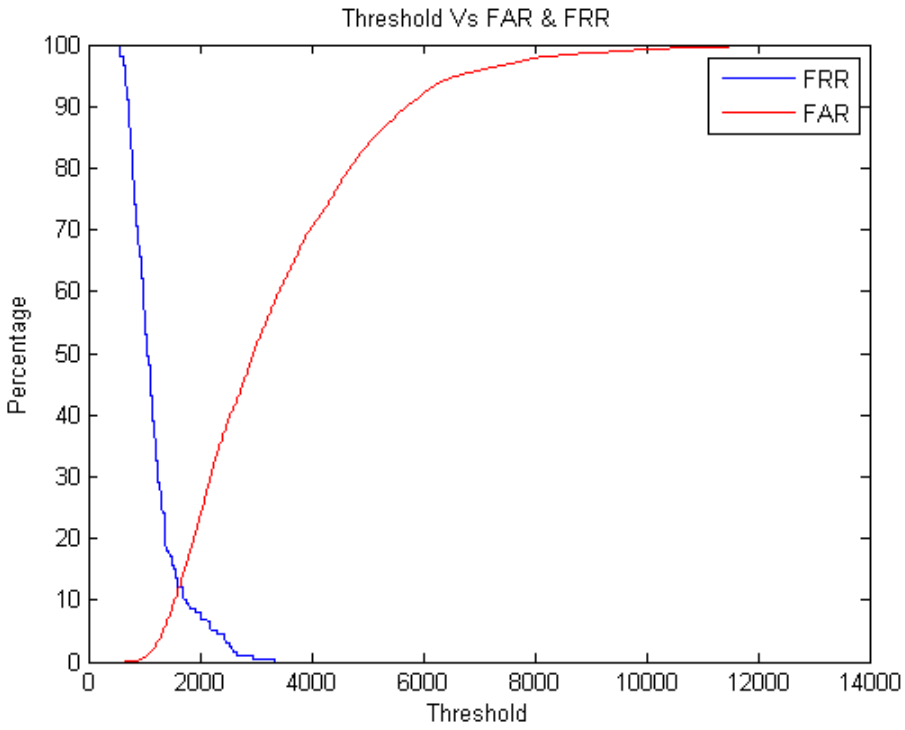

Figure 5.127: EER of 480 Zernike features tested on DB4

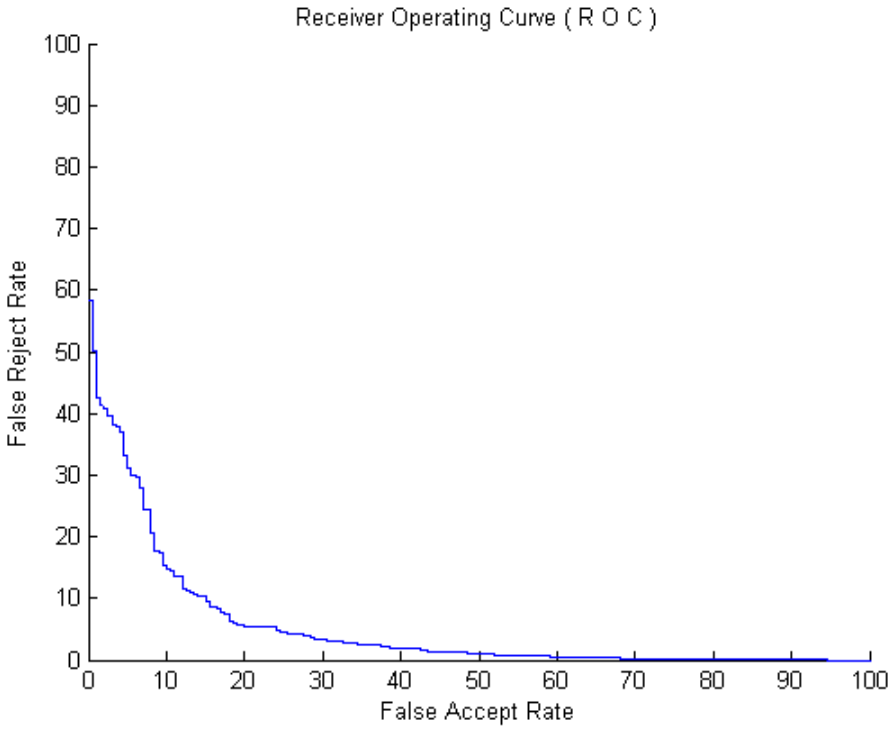

Figure 5.128: ROC Curve of 480 Zernike features tested on DB4

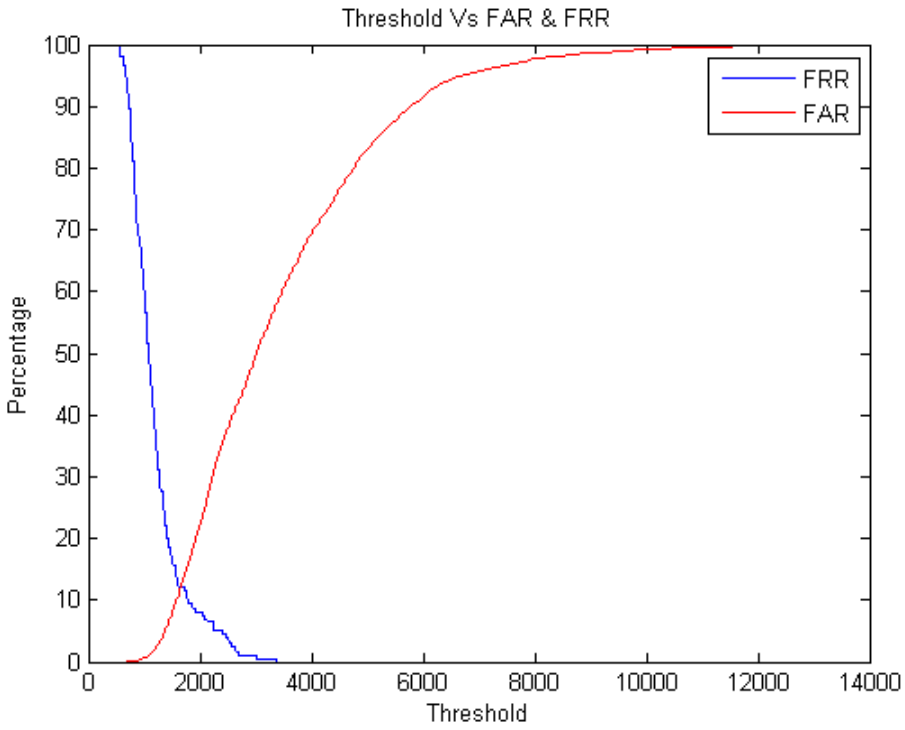

Figure 5.129: EER of 490 Zernike features tested on DB4

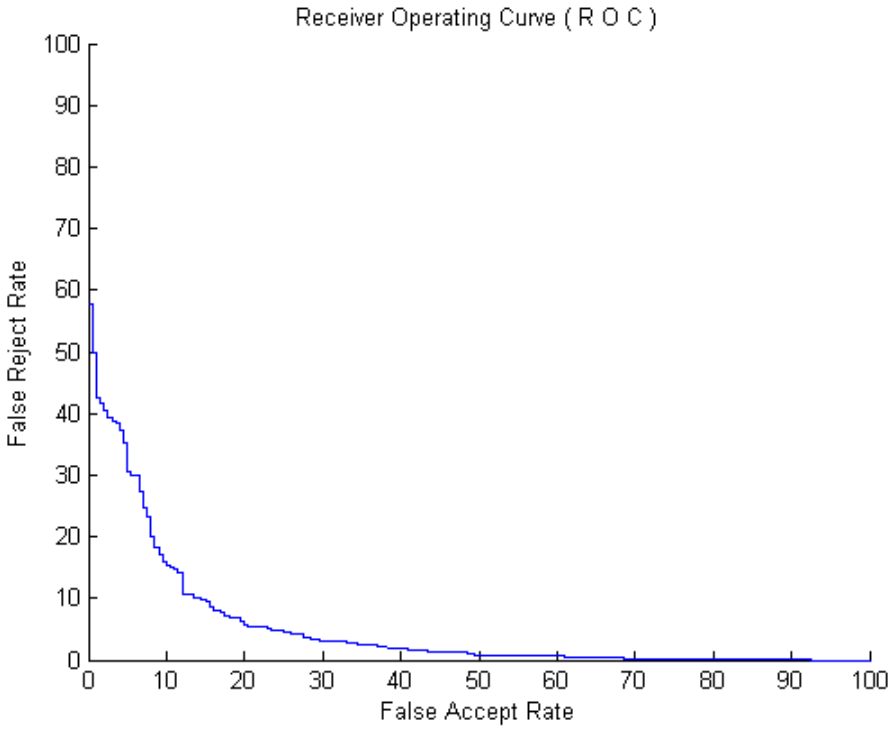

Figure 5.130: ROC Curve of 490 Zernike features tested on DB4

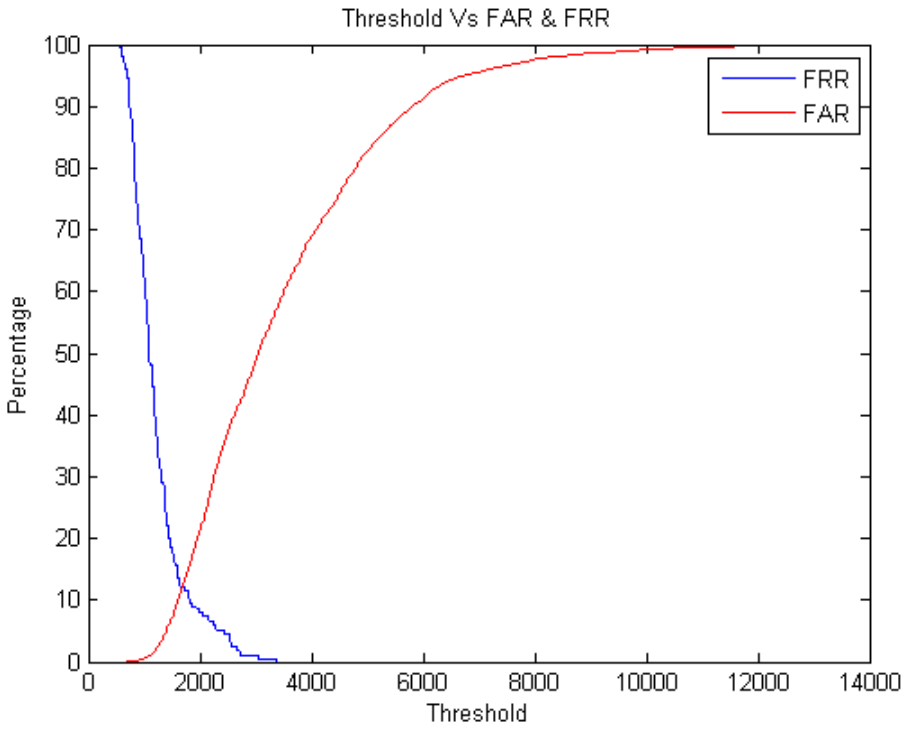

Figure 5.131: EER of 500 Zernike features tested on DB4

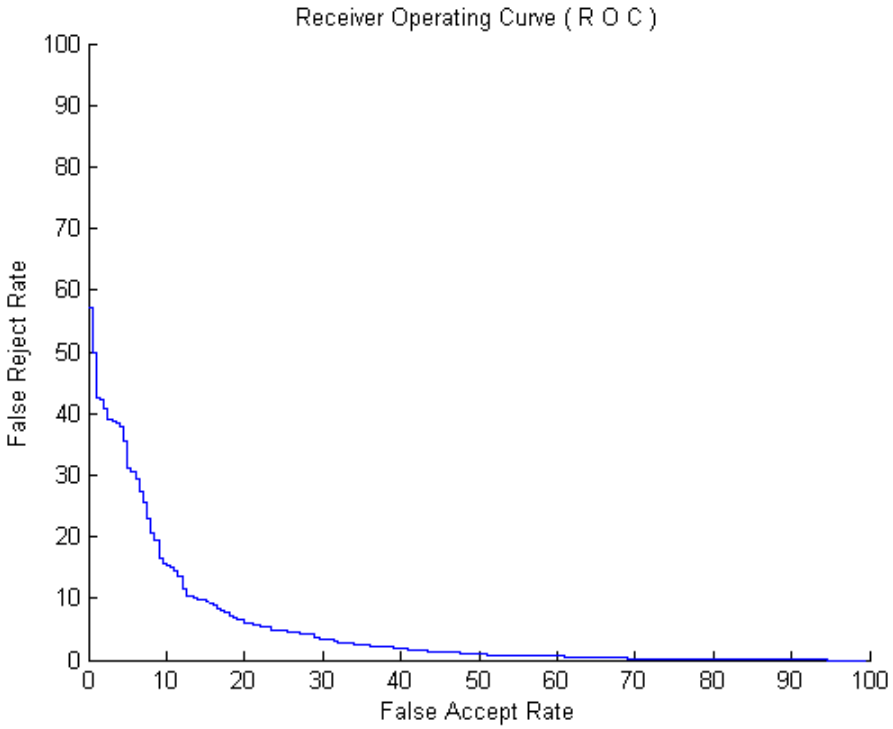

Figure 5.132: ROC Curve of 500 Zernike features tested on DB4

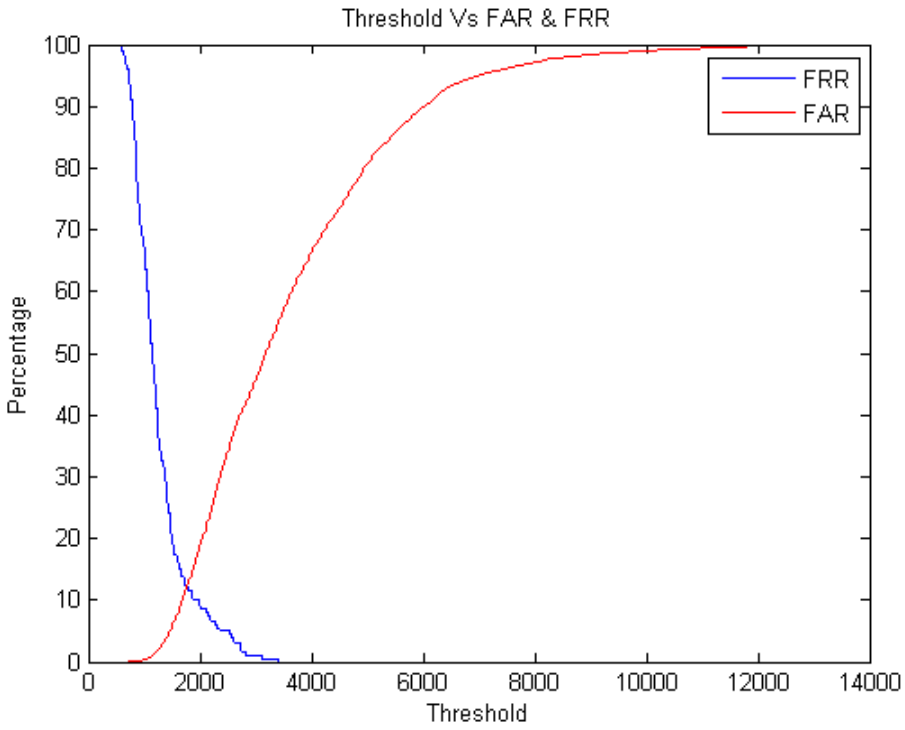

Figure 5.133: EER of 520 Zernike features tested on DB4

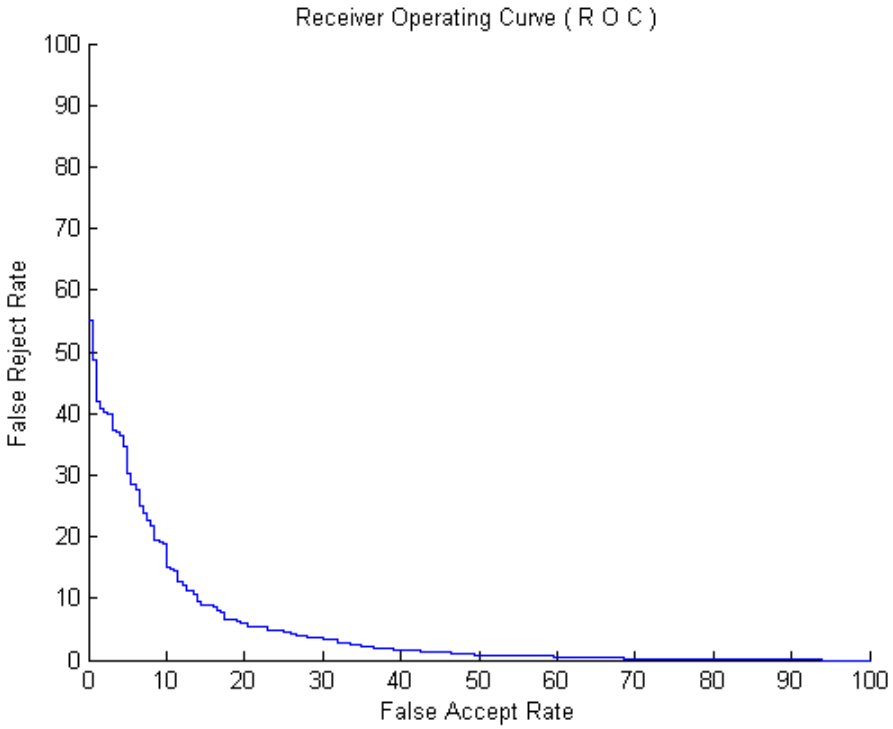

Figure 5.134: ROC Curve of 520 Zernike features tested on DB4

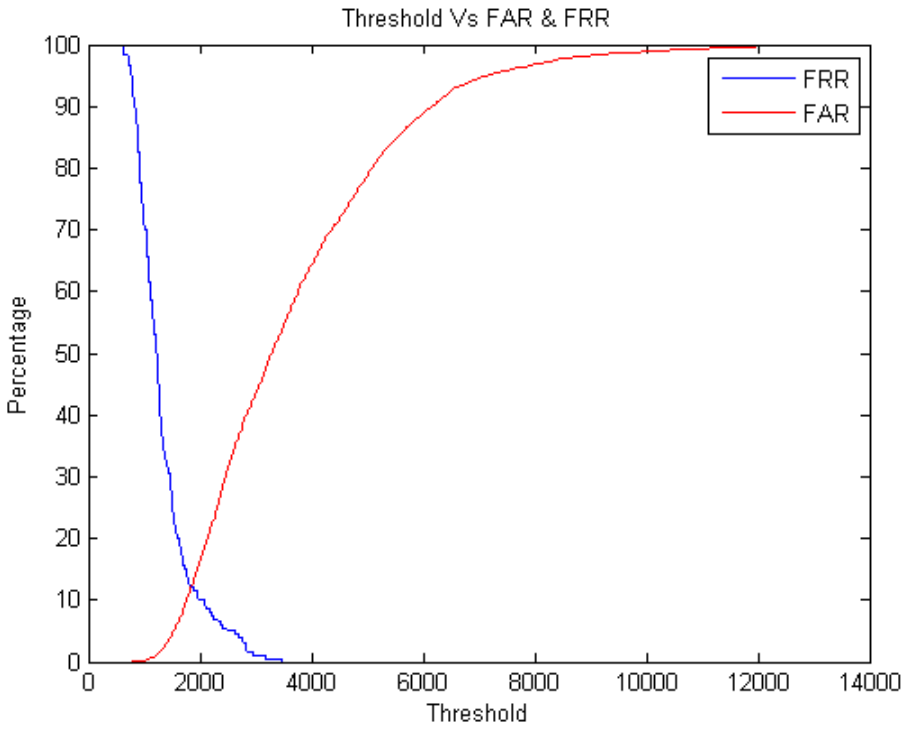

Figure 5.135: EER of 550 Zernike features tested on DB4

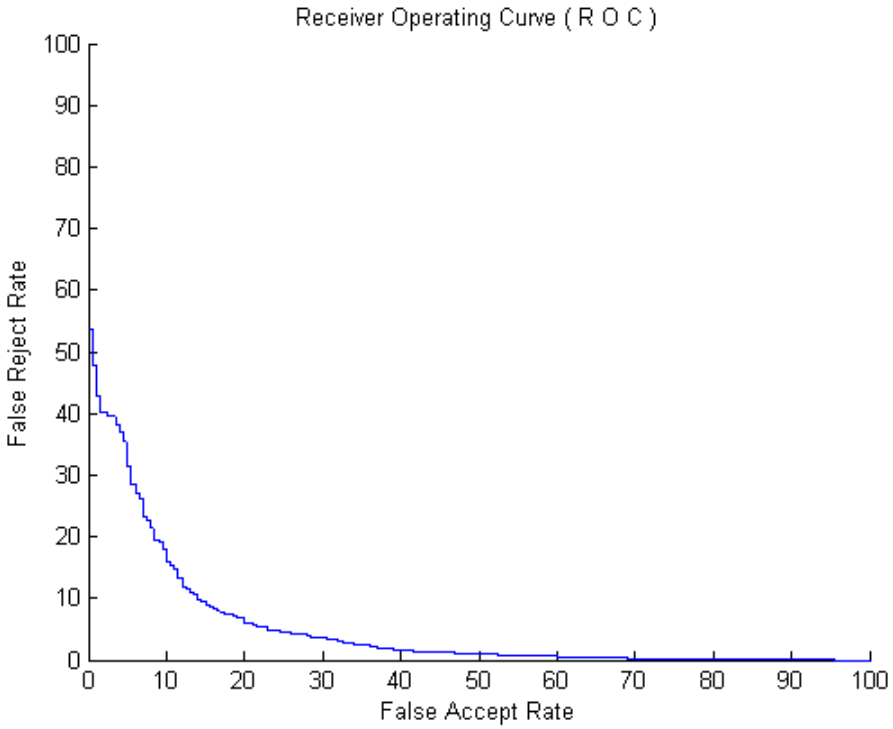

Figure 5.136: ROC Curve of 550 Zernike features tested on DB4

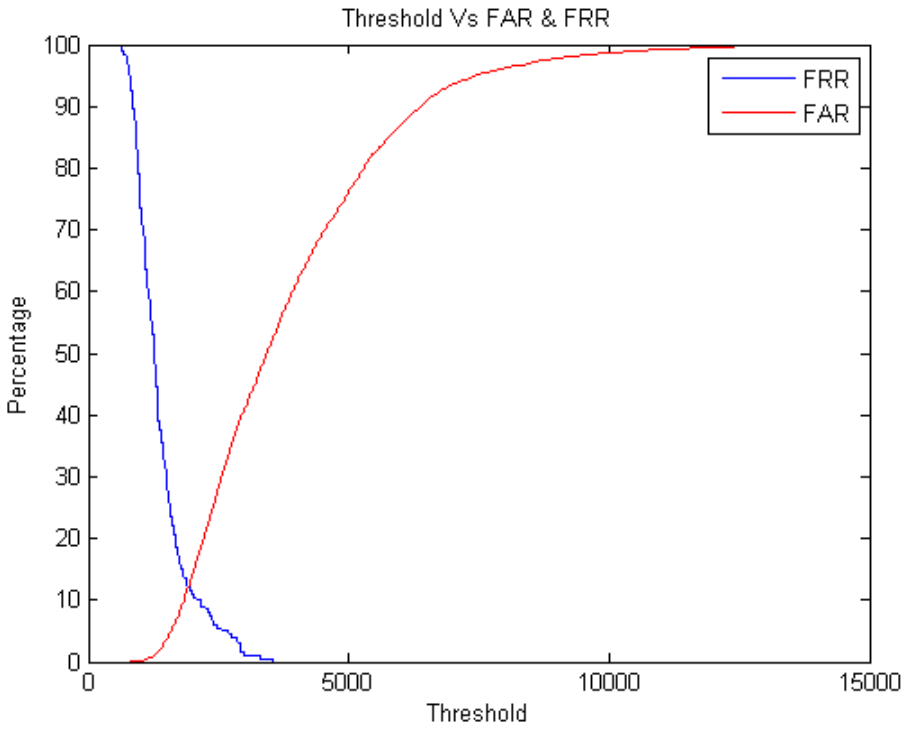

Figure 5.137: EER of 570 Zernike features tested on DB4

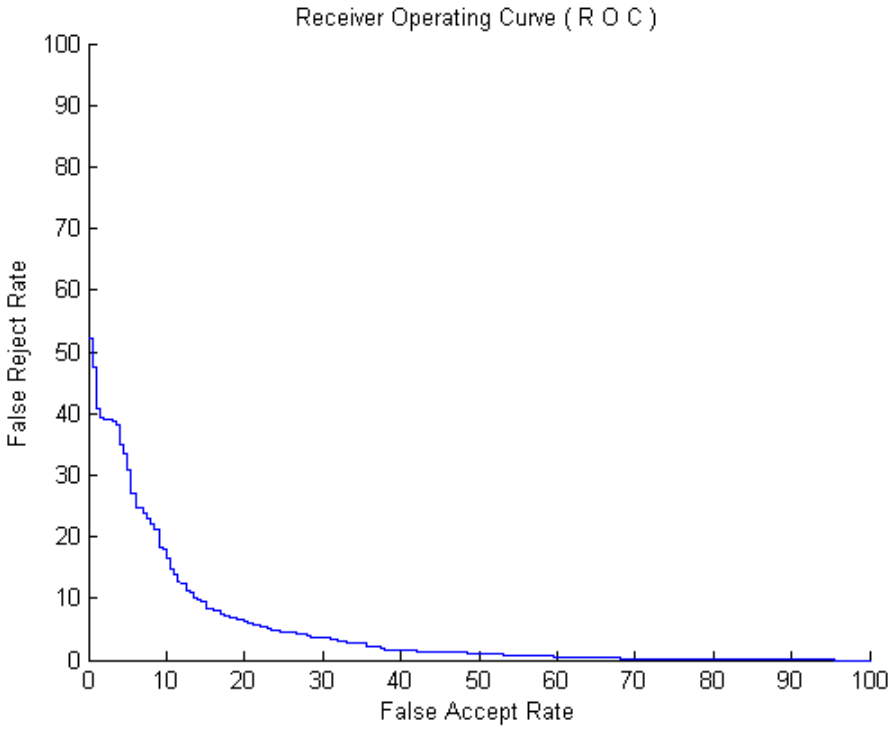

Figure 5.138: ROC Curve of 570 Zernike features tested on DB4

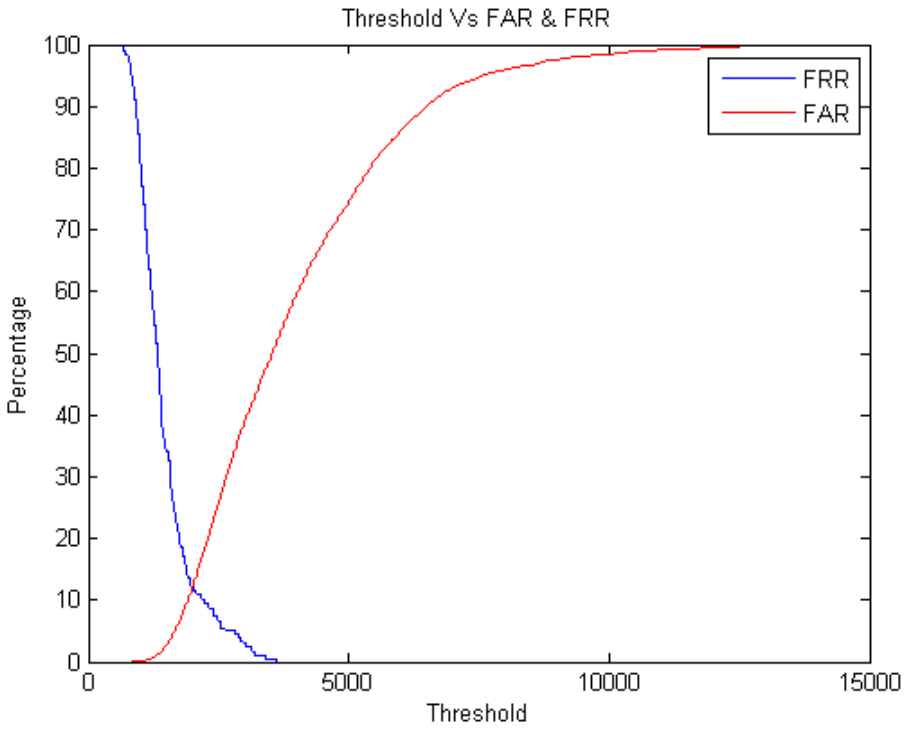

Figure 5.139: EER of 600 Zernike features tested on DB4

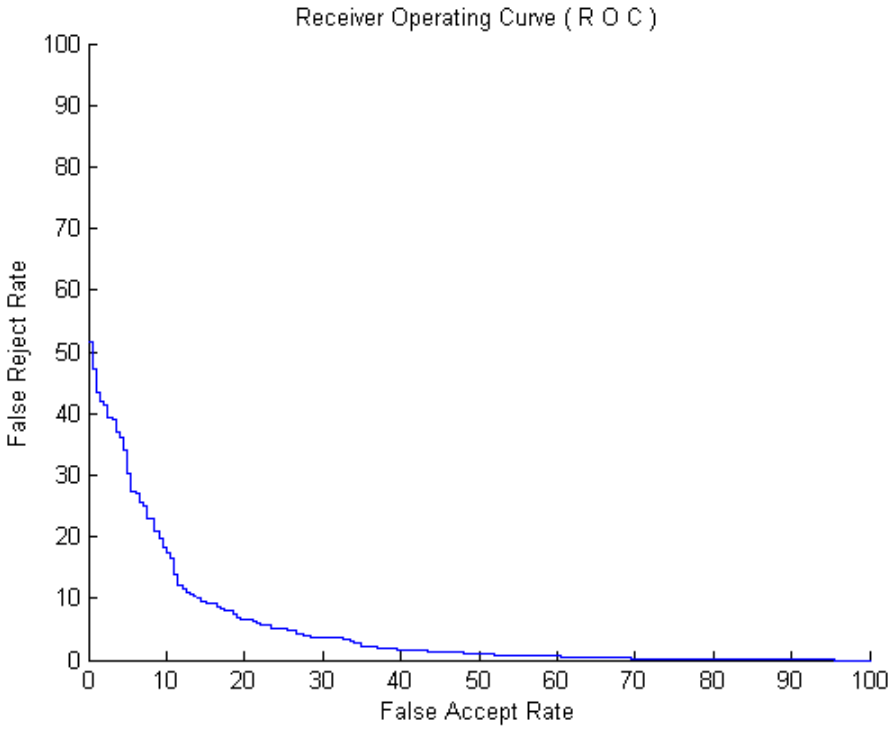

Figure 5.140: ROC Curve of 600 Zernike features tested on DB4

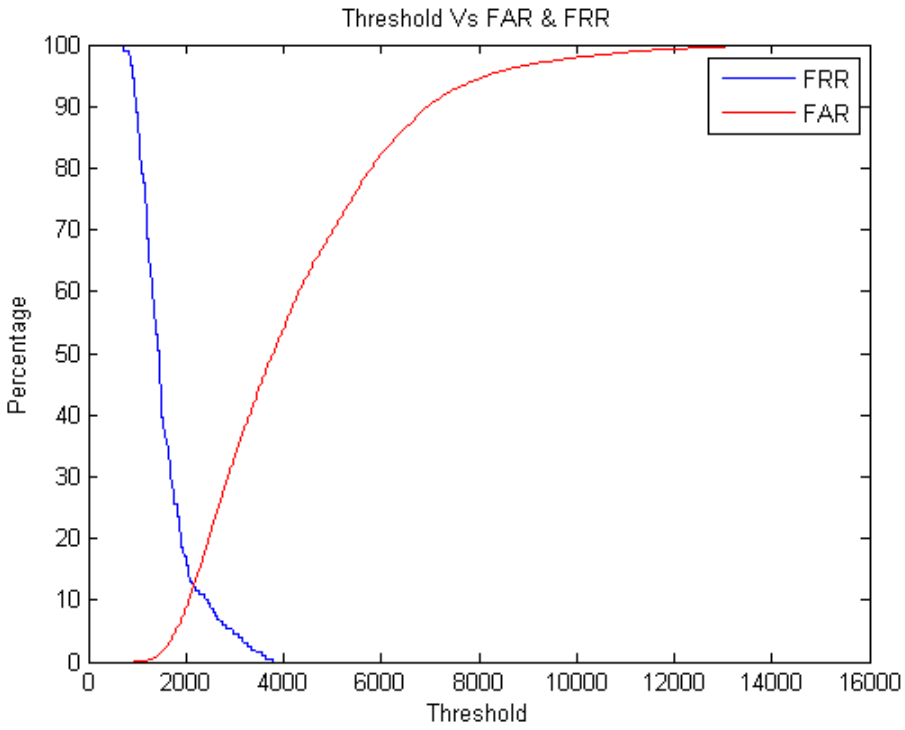

Figure 5.141: EER of 650 Zernike features tested on DB4

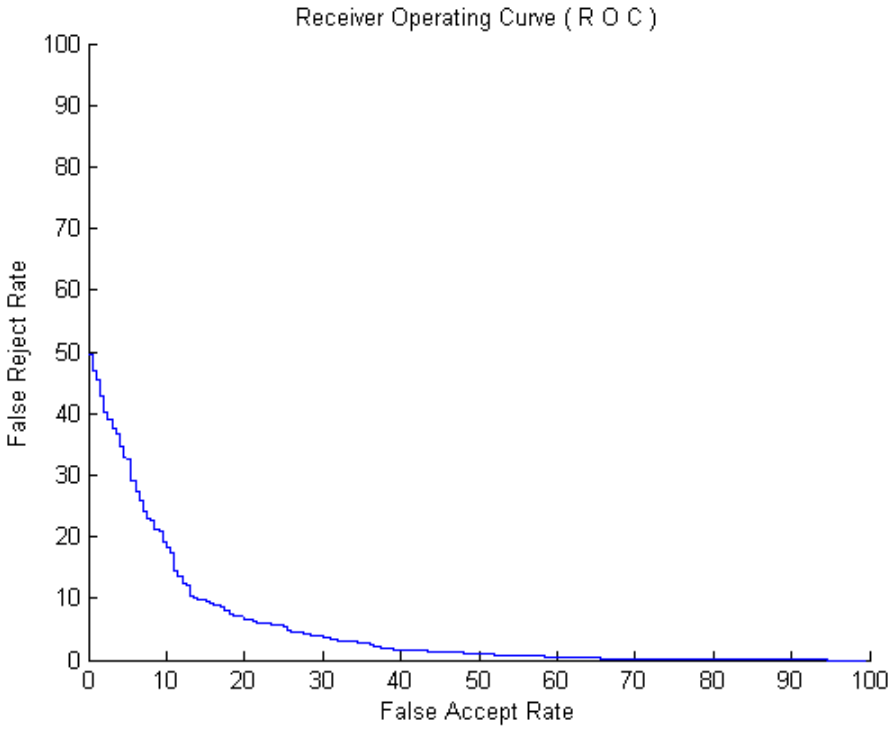

Figure 5.142: ROC Curve of 650 Zernike features tested on DB4

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this dissertation, a novel hierarchical k-means clustering based fingerprint quality classification method and a novel fingerprint matching method based on Zernike moments has been proposed. In a hierarchical k-means clustering based fingerprint quality classification method, a set of Fourier and statistical features have been calculated. A hierarchical k-means clustering algorithm has been utilized to classify the fingerprint image into one of four quality classes i.e. good, dry, normal or wet. These centroid values of clusters have been used to classify the fingerprint image as good, wet, dry or normal quality. An objective method has also been proposed to evaluate the performance of fingerprint quality classification. It has been shown through experimental results that the performance of minutiae based matcher improves when the quality of fingerprint image is incorporated in the matching stage. The false accept rate and false reject rate of minutiae based fingerprint matcher is 1.8 on FVC 2002 db1 database without utilizing fingerprint quality information. False accept rate has been reduced from 1.8 to 0.79 whereas the false reject rate is at 1.8 when fingerprint quality based threshold value is utilized. This significant improvement in the performance of fingerprint matching system shows the effectiveness of hierarchical k-means clustering technique in quality based classification of fingerprints.

We have also proposed a novel fingerprint matching technique using Zernike moments. For fingerprint matching, it is desirable to obtain a fingerprint representation invariant to translation and rotation. Translation invariance is achieved by transforming the fingerprint image into frequency domain and taking the absolute yielding the spectrum of an image invariant to translation. For rotation invariance, Zernike moments are calculated which are invariant to rotation. The fingerprint image is first enhanced and then converted into frequency domain by taking its Discrete Fourier Transform. Then the magnitude of the Zernike moments is calculated. The fingerprint matching is based on the normalized Euclidean distance between the two corresponding Zernike moments of stored template and query fingerprint image. Different number of Zernike features has been tested on FVC 2002 Db1,Db2, Db3 and Db4 databases. The number of Zernike features used in fingerprint matching varies for each database to obtain the minimum EER. In our proposed algorithm, translation is handled by taking the magnitude of the DFT of the fingerprint image which is translation invariant. Zernike moments exhibit rotation invariance property. Therefore Zernike features extracted were rotationally invariant. The performance of our proposed algorithm is very good as it performed better as compare to other matching schemes as shown by experimental results discussed in chapter 5.

6.2 Future Work

For fingerprint quality classification, further works need to be explored to enhance the classification accuracy of the proposed approach. To improve the classification performance, different types of other frequency or statistical features can also be explored. This technique can be combined with other quality estimation approaches to enhance the performance of the fingerprint matching system.

For Zernike moments based fingerprint matching, further works need to be explored to reduce the feature extraction time. To improve the EER, different types of other moments can be explored and this technique can be combined with other feature based matching techniques to enhance the performance.

The following areas of improvement are suggested:

i. Extraction of features by using adaptive methods for optimal selection of the feature set.

ii. By improving the accuracy of registration of fingerprint image, a better performance can easily be expected.

iii. By exploring more discriminative features to make a feature set, a better performance can be achieved.

iv. By optimizing the algorithm for calculation of Zernike Moments, the total execution time can be improved.

v. To increase verification accuracy in matching, two different fingers (thumb and right index) of a person can be used separately in verification. None of the two fingers of same person are similar. This should improve verification accuracy.

vi. In military application where security is very high, verification can be done using all the fingers of one hand. This means that 5 fingers are used in matching and this increases the accuracy of biometric system.

Annexure A

List of Abbreviations

Abbreviations	Meanings
db1	Database 1
db2	Database 2
db3	Database 3
db4	Database 4
EER	Equal Error Rate
FAR	False Accept Rate
FRR	False Reject Rate
FVC 2002	Fingerprint Verification Competition 2002
GAR	Genuine Accept Rate
GRR	Genuine Reject Rate
ROC	Receiver Operating Curve
ROI	Region of Interest

References

[1]. J. L. Wayman, “Fundamentals of Biometric Authentication Technologies,” International Journal of Imaging and Graphics 1(1) (2001)

[2]. A.K. Jain and S. Pankanti, “Automated Fingerprint Identification and Imaging Systems,” in Advances in Fingerprint Technology, 2nd ed. New York: Elsevier Science, 2001.

[3]. A. K. Jain, A. Ross and S. Prabhakar, “An Introduction to Biometric Recognition”, IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics, Vol. 14, No. 1, pp. 4-20, January 2004

[4]. A. K. Jain, S. Prabhakar, and S. Pankanti, “On the similarity of identical twin fingerprints”, Pattern Recognition, vol. 35, no. 8, pp. 2653-2663, 2002

[5]. S. Pankanti, S. Prabhakar, and A. K. Jain, “On the individuality of fingerprints", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 8, pp. 1010-1025, 2002

[6]. A. Jain, L. Hong, S. Pankanti, and R. Bolle. On-line identity-authentication system using fingerprints. Proceedings of IEEE (Special Issue on Automated Biometrics), 85(9):1365–1388, September 1997

[7]. L. Hong, "Automatic Personal Identification Using Fingerprints", PhD Thesis, Michigan State University, 1998

[8]. A. K. Jain, L. Hong, and S. Pankanti. Biometric identification. Comm. ACM, pages 91–98, Feb 2000.

[9]. A.K. Jain, L. Hong and R. Bolle, "On-line Fingerprint Verification", IEEE Transactions on PAMI, Vol. 19, No. 4, pp. 302-314, 1997.

[10]. S. Prabhakar. "Fingerprint Classification and Matching Using a Filterbank", PhD Thesis, Michigan State University, 2001.

[11]. F. Galton, “Finger-Prints”, 1892

[12]. F. Galton, “Finger-Print Directories”, 1895

[13]. E. Henry, “The Classification and Uses of Finger Prints”, December 1900.

[14]. Access Control Applications using Optical Computing. http://www.mytec.com/, 1997.

[15]. N. Ratha, S. Chen, and A. K. Jain, “Adaptive Flow Orientation-Based Feature Extraction in Fingerprint Images”, Pattern Recognition, Vol. 28, No.11, pp. 1657-1672, 1995.

[16]. R. W. Frischholz and U. Dieckmann, “Bioid: A Multimodal Biometric Identification System”, IEEE Computer, Vol. 33, No. 2, pp. 64-68, 2000.

[17]. S. Gold and A. Rangarajan, “A graduated assignment algorithm for graph matching”, IEEE Trans. Pattern Anal. Machine Intell., vol. 18, no. 4, pp. 377–388, 1996.

[18]. D. Maio and D. Maltoni, “Direct gray-scale minutiae detection in fingerprints,"IEEE Transactions on PAMI, vol. 19, pp. 27-40, Jan 1997.

[19]. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer Verlag, June 2003.

[20]. A. K. Jain, A. Ross, and S. Prabhakar, “Fingerprint matching using minutiae and texture features," in Proc. International Conference on Image Processing (ICIP), (Thessaloniki, Greece), pp. 282{285, Oct 2001.

[21]. A.K. Jain, S. Prabhakar, L. Hong and S. Pankanti, "FingerCode: A Filterbank for Fingerprint Representation and Matching", *Proc. IEEE Conference on CVPR*, Colorado, Vol. 2, pp. 187-193, June 23-25, 1999.

[22]. A. K. Jain, S. Prabhakar, L. Hong and S. Pankanti "Filterbank-based Fingerprint Matching", *IEEE Transactions on Image Processing*, Vol. 9, No.5, pp. 846-859, May 2000.

[23]. A. Ross, J. Reisman, and A. K. Jain, Fingerprint Matching Using Feature Space Correlation. Proc. of Post-ECCV Workshop on Biometric Authentication, LNCS 2359, pp.48-57, Denmark, 2002.

[24]. K. Ito, H. Nakajima, K. Kobayashi, T. A.T. Higuchi A Fingerprint Matching Algorithm Using Phase-Only Correlation, IEICE Trans, Fundamentals, Vol. E87-A, No.3, March 2004.

[25]. C.D. Kuglin and D.C. Hines, "The phase correlation image alignment method", Proc. Int. Conf. on Cybernetics and Society, pp. 163-165, 1975.

[26]. T. Kenji, T. Aoki, Y. Sasaki, T. Higuchi, and K. Kobayashi, "High-accuracy subpixel image registration based on phase-only correlation", IEICE Trans. Fundamentals, vol.E86-A, no.8, pp.1925-1934, Aug. 2003.

[27]. A. Ross, A. K. Jain, J. Reisman, A Hybrid Fingerprint Matcher, *Pattern Recognit.* 36 (7) (2003) 1661-1673.

[28]. E. K. Yun, S.b. Cho, Adaptive Fingerprint Image Enhancement with Fingerprint Image Quality Analysis, Image Vis. Comput. 24 (2006) 101–110.

[29]. E. Lim, X. Jiang, W. Yau, Fingerprint Quality and Validity analysis, in: Proceedings of IEEE International Conference on Image Processing, 1 (2002) 469-472.

[30]. Y. Chen, S. C. Dass, A.K. Jain, Fingerprint Quality Indices for Predicting Authentication Performance, in: Proceedings of Audio Video based Biometric Person Authentication, Springer, LNCS 3546, (2005) 160-170.

[31]. L. Hong, Y. Wan, A.K. Jain, Fingerprint Image Enhancement: Algorithm and Performance Evaluation, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 777–789.

[32]. L.L. Shen, A. Kot, W.M. Koo, Quality Measures of Fingerprint Images, in: Proceedings of Audio Video based Biometric Person Authentication, Springer, LNCS 2091, (2001) 182–271.

[33]. R.C. Gonzalez, R.E. Woods, Digital Image Processing, 2nd ed., Prentice-Hall, (2002) 154-155.

[34]. R.C. Gonzalez, R.E. Woods, Digital Image Processing, 2nd ed., Prentice-Hall, (2002) 151-152.

[35]. W. K. Pratt, Digital Image Processing: PIKS Inside, 3rd ed., John Wiley & Sons, (2001) 448-469.

[36]. C. Wu, Advanced Feature Extraction Algorithms for Automatic Fingerprint Recognitio Systems, PhD Thesis, The State University of New York at Buffalo, (2007) 37-46.

[37]. W. K. Pratt, Digital Image Processing: PIKS Inside, 3rd ed., John Wiley & Sons, (2001) 511-515.

[38]. E. Gose, R. Johnsonbaugh, S. Jost, Pattern Recognition and Image Analysis, Prentice Hall, (1996) 213-214.

[39]. G. Gan, C. Ma, J. Wu, Data Clustering: Theory, Algorithms and Applications, Society for Industrial Mathematics, (2007) 161-164.

[40]. T. Jea, Minutiae Based Partial Fingerprint Recognition , PhD Thesis, The State University of New York at Buffalo, (2005) 33-68.

[41]. D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition, Springer Verlag, (2003) 13-19.

[42]. J. C. Yang, D. S. Park, A Fingerprint Verification Algorithm using Tessellated Invariant Moment Features, Neurocomputing 71 (2008) 1939– 1946.

[43]. D. Maio, L. Nanni, An Efficient Fingerprint Verification System using Integrated GaborFilters and Parzen Window Classifier, Neurocomputing 68 (2005) 208–216.

[44]. D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition, Springer Verlag, (2003) 75-76.

[45]. D. Maio, D. Maltoni, R. Cappelli, J. L.Wayman, A. K. Jain, FVC2002: Second Fingerprint Verification Competition, in: Proceedings of International Conference on Pattern Recognition, (2002) 811-814.

[46]. A. K. Jain, S. Prabhakar, L. Hong and S. Pankanti "Filterbank-based Fingerprint Matching", IEEE Trans. on Image Proc. 9 (5) (2000) 846-859.

[47]. A. Khotanzad, Y.H. Hong, Invariant image recognition by Zernike moments, IEEE Trans. Pattern Anal. Mach. Intell. 12 (5) (1990) 489–497.

[48]. M. R. Teague, "Image analysis via the general theory of moments", J. Opt. Soc. Amer. 70 (1980) 920-930.

[49]. M.K. Hu, Visual pattern recognition by moment invariants, IRE Trans. Info. Theory IT-8 (1962) 179–187.

[50]. R. Mukundan, “Image analysis by Tchebichef moments”, IEEE Trans. on Image Proc. 10 (9) (2001) 1357-1364.

[51]. C.-H. The, R.T. Chin, On image analysis by the method of moments, IEEE Trans. Pattern Anal. Mach. Intell. 10 (4) (1988) 496–513.

[52]. F. Zernike, Physica, vol. 1, p. 689, 1934.

[53]. D. Sim, H. Kim, R. Park, Invariant texture retrieval using modified Zernike moments, Image Vis. Comput. 22 (2004) 331–342.

[54]. A.K. Jain, L. Hong, S. Pankanti and R. Bolle, An Identity Authentication System Using Fingerprints, Proc. IEEE, 85 (9) (1997) 1365-1388.

[55]. F. Benhammadi, M.N. Amirouche, H. Hentous, K.B. Beghdad, M. Aissani, Fingerprint matching from minutiae texture maps, Pattern Recognit. 40 (1) (2007) 189–197.

[56]. J. Liu, Z. Huang, K. Chan, Direct minutiae extraction from gray-level fingerprint image by relationship examination, International Conference on Image Proc. 2 (2000) 427–430.

[57]. J. C. Yang, D. S. Park, A fingerprint verification algorithm using tessellated invariant moment features, Neurocomputing 71 (2008) 1939– 1946.

[58]. L. Wang, G. Healey, Using Zernike moments for the illumination and geometry invariant classification of multispectral texture, IEEE Trans. on Image Proc. 7 (2) (1998) 196–203.

[59]. D. Maio, L. Nanni, An efficient fingerprint verification system using integrated gabor filters and Parzen Window Classifier, Neurocomputing, 68 (2005) 208–216.

[60]. T. Amornraksa, S. Tachaphetpiboon, Fingerprint recognition using DCT features, Electron. Lett. 42 (9) (2006) 522–523.

[61]. A.T.B. Jin, D.N.C. Ling, O.T. Song, An efficient fingerprint verification system using integrated wavelet and Fourier-Mellin invariant transform, Image Vis. Comput. 22 (6) (2004) 503–513.

[62]. M. Tico, P. Kuosmanen, J. Saarinen, Wavelet domain features for fingerprint recognition, Electron. Lett. 37 (1) (2001) 21–22.

[63]. A. Ross, A. K. Jain, and J. Reisman, "A Hybrid Fingerprint Matcher", Pattern Recognit. 36 (7) (2003) 1661-1673.

[64]. L. Nanni, A. Lumini, A hybrid wavelet-based fingerprint matcher, Pattern Recognit. 40 (11) (2007) 3146–3151.

[65]. A. K. Jain, A. Ross, and S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features", Proc. International Conference on Image Proc.(ICIP), (2001) 282-285, Greece.

[66]. L. Hong, Y. Wan, and A.K. Jain, "Fingerprint Image Enhancement: Algorithms and Performance Evaluation", IEEE Trans. Pattern Anal. Mach. Intell. 20(8) (1998) 777-789.

[67]. R.C. Gonzalez, R.E. Woods, Digital Image Processing, 2nd ed., Prentice-Hall, (2002) 154-155.

[68]. W. K. Pratt, Digital Image Processing: PIKS Inside, 3rd ed., John Wiley & Sons, (2001) 516-517.

[69]. http://bias.csr.unibo.it/fvc2002/databases.asp

[70]. D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer Verlag, (2003) 96-100.

[71]. A.K. Jain, L. Hong and R. Bolle, "On-line Fingerprint Verification", IEEE Trans. Pattern Anal. Mach. Intell. 19 (4) (1997)302-314.

[72]. L.Nanni, A.Lumini, A novel method for fingerprint verification that approaches the problem as a two-class pattern recognition problem, Neurocomputing, 69 (2006) 846–849.

[73]. M. U. Munir, M. Y. Javed, "Fingerprint Matching using Ridge Patterns", Proceedings of the 1st International Conference on Information & Communication Technologies (ICICT), pp. 116-120, Karachi, August 27-28, 2005.

[74]. M. U. Munir, M. Y. Javed, "Ridge Feature based Fingerprint Verification", Proceedings of National Conference on Information Technology and Applications, pp. 56-63, Quetta, Pakistan, April 21- 22, 2005

[75]. M. U. Munir, M. Y. Javed , "Fingerprint Matching using Gabor Filters", Proceedings of the National Conference on Emerging Technologies (NCET), pp. 147-151, Karachi, December 18-19, 2004.

[76]. F.A. Fernandez, J Fierrez, J. O. Garcia, J. G. Rodriguez, H. Fronthaler, K. Kollreider, J. Bigun, "A Comparative Study of Fingerprint Image-Quality Estimation Methods", IEEE Transactions on Information Forensics and security, Vol. 2, No. 4, 2007, pp. 734-743.

[77]. E.Lim, K. Toh, P. Suganthan, X. Jiang, and W. Yau, "Fingerprint image quality analysis", in Proc, International Conference on Image Processing, 2004,pp. 1241-1244.

[78]. M. U. Munir, M. Y. Javed, S. A. Khan, "A Hierarchical K-means Clustering based Fingerprint Quality Classification", Neurocomputing, 2012

[79]. H. A. Qader, A. R. Ramli, S. Al-Haddad, " Fingerprint Recognition Using Zernike Moments", The International Arab Journal of Information Technology, Vol. 4, No. 4, 2007, pp. 372-376.