Minutiae Triplet-based Features with Extended Ridge Information for Determining Sufficiency in Fingerprints

Kevin Hoyle

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Michael S. Hsiao, Chair
Lynn Abbott
Edward A. Fox

July 6, 2011
Blacksburg, VA

Keywords: Fingerprints, Quality, Latent, Sufficiency, Minutia, Friction Ridges, Triangles, Triplets

Minutiae Triplet-based Features with Extended Ridge Information for Determining Sufficiency in Fingerprints

Kevin Hoyle

ABSTRACT

In order to deliver statistical and qualitative backing to latent fingerprint evidence, algorithms are proposed (1) to perform fingerprint matching to aid in quality assessment, and (2) to discover statistically rare features or patterns in fingerprints. These features would help establish an objective minimum-quality baseline for latent prints as well as aid in the latent examination process in making a matching comparison. The proposed methodologies use minutiae triplet-based features in a hierarchical fashion, where not only minutia points are used, but ridge information is used to help establish relations between minutiae. Results show (1) that our triplet-based descriptor is useful in eliminating false matches in the matching algorithm, and (2) that a set of distinctive features can be found that have sufficient discriminatory power to aid in quality assessment.

Acknowledgements

I would first like to thank my research advisor Dr. Michael Hsiao for giving me this research opportunity and for all his hard work and guidance throughout this research. I learned a lot from him starting from the Testing and Verification courses through the writing of this thesis. I also thank my committee members Dr. Lynn Abbott and Dr. Ed Fox for their ideas and advice throughout the past year and for serving on my committee. Lastly I want to thank the friends I made in Blacksburg and the Virginia Tech community as a whole for making my time here fun and rewarding.

Contents

List of Figures

List of Tables

Chapter 1: **Introduction**

Since 1893, when the Home Ministry Office, UK, accepted that no two individuals have the same fingerprints, law enforcement departments have been using fingerprints to identify repeat criminals [1]. In the early twentieth century, fingerprint recognition was formally accepted as a valid personal identification method and became a standard routine in forensics [2]. By matching leftover fingerprint smudges (latents) from crime scenes to fingerprints collected during booking, authorities could determine the identity of criminals who had been previously arrested. Due to the painstaking process of indexing and matching the millions of fingerprints that exist in modern databases, AFIS, or Automated Fingerprint Identification Systems, have been in wide use over the past four decades [1].

It wasn't until relatively recent that court cases emerged that challenged the legality of using fingerprint identification as evidence. From [3], "Trial judges have rejected the dozen challenges filed since 1999, holding that fingerprinting, which has been accepted since 1911, has proved its reliability in the courtroom." Defense lawyers are targeting the fingerprint identification process' lack of scientific backing to get fingerprint evidence declared inadmissible in court. Fingerprint evidence has been challenged by a couple of major cases in particular — Daubert v. Merrell Dow Pharmaceuticals in 1993 and Kumho Tire Company v. Carmichael in 1999 — in which the Supreme Court declared that federal judges must determine the reliability of expert testimony before admitting it [3]. In order to improve the ability of fingerprint evidence in court, in 1999 the Department of Justice's research arm, the National Institute of Justice, put out a call for studies to come up with standardized, statistically tested procedures for comparing fingerprints that produce correct results with acceptable error rates. This solicitation was largely based on an earlier report by the NIJ [4], within which specific needs for latent print examinations were identified.

Latent fingerprint matching poses the greatest problem for Automated Fingerprint Identification Systems (AFIS), as well as for use as evidence in a courtroom. Latent fingerprints are typically those obtained from a crime scene, and are often partial and/or of poor quality. Dealing with only partial prints and poor quality is the main challenge for AFIS and in trying to deliver a courtroom decision of *identification* or *exclusion*.

Friction ridge patterns on human fingers have been generally accepted as being unique to each individual person; that is, it is accepted that every person has a unique set of patterns as determined by the presence and arrangement of the friction ridges, composing specific features [5-8]. However, a proven mathematical basis or statistical approach has not been established to quantify the rarity of such patterns. Nor can we scientifically claim friction ridge uniqueness or individualization.

Attempts have been made to determine fingerprint individuality, which refers to the premise that a fingerprint is unique to a particular person [8, 9]. This premise is scientifically unproven, but has come to be accepted over time, mainly because of the lack of counterexamples. Fingerprint individuality, in essence, refers to the distinctiveness of fingerprints from different sources. Due to distortion and noise, impressions from the same fingertip will differ. Both human examiners and AFIS determine that two prints originated from the same finger if they are “sufficiently” similar. How much similarity is sufficient is dependent upon the amount of intra-class variation present in prints originating from the same finger. Fingerprint individuality can be formulated in three ways, according to Pankanti et al. [8], depending on what is under examination: 1) determine whether any two (or more) fingerprints from a given population are sufficiently similar, 2) “given a sample fingerprint, determine the probability of finding a sufficiently similar fingerprint in a target population”, and 3) “given two fingerprints from two different sources, determine the probability that they are sufficiently similar.” Solutions for formulations 1 and 2 and can be derived from the solution for formulation 3. *Individuality models* are developed to attempt to solve these formulations of individuality.

1.1 Motivation

There is a need to analyze latent print quality to determine if there is sufficient information and/or sufficient quality present in a latent print to proceed with an examination, comparison, or claim identification or exclusion. In this work we identify and quantify distinctive features that have high discriminatory power found in fingerprints, and we attempt to achieve multiple goals: (1) solidify the scientific foundation for assessment of quality, or lack thereof, for latent fingerprint images; (2) develop a more complete set of features from which to select descriptors; and (3) develop objective metrics to determine the extent to which a given latent print is of sufficient quality to proceed with an examination, comparison, and identification or exclusion.

These goals are motivated by the need for the validation of the science behind fingerprint identification methods. This need has been expressed by the NIJ in [4, 10] and by the National Academy of Sciences in [7]. The NIJ expected the research it was soliciting to “address the relative importance of different minutiae to establish individuality, as well as the statistical significance of groups of minutiae” [10]. In [4], one of the stated needs is “Determination of the relative importance of print features.” With that knowledge it may allow examiners to quantify the significance of a match between fingerprints in a more definitive way. This also may lead to improved productivity for examiners. We address these objectives using extended attributes of minutiae triplets.

1.2 Contributions of this Thesis

This thesis makes the following contributions to the fingerprint sufficiency study:

- Development of a modular, hierarchical framework for forming minutia triangle features
- Incorporating ridge information into the minutia triangle features
- Development of a minutia triangle feature-based matching algorithm for use in quality assessment and human examiner verification/support
- Development of an algorithm for finding distinctive spatial distributions of minutia triangle clusters to provide scientific backing to the assessment of quality and help establish an objective minimum-quality baseline for latent prints

1.3 Outline

Chapter 2 lays the background for biometric matching systems; fingerprint image capture for tenprint and latent images; fingerprint features; automated fingerprint matching; and individuality and quality sufficiency of fingerprints. Chapter 3 describes the contributions in developing the triangle-based features and the matching algorithm. In Chapter 4, the algorithm for finding distinctive spatial distributions of minutiae triangle clusters is presented. The conclusion is given as Chapter 5.

Chapter 2: **Background**

2.1 Biometric Systems

Depending on the application context, a biometric system is classified as a *verification* system or an *identification* system.

A verification system authenticates a person's identity by comparing the captured biometric characteristic with the previously known (enrolled) reference template pre-stored in the system. It conducts a one-to-one comparison to confirm or deny the submitted claim of identity by the individual. For example, a fingerprint reader scans a fingerprint to determine whether the user is the appropriate person. The fingerprint is processed by a feature extraction module to produce a feature set. That feature set is fed to a matcher, where it is compared to the known enrolled fingerprint of the user.

An identification system attempts to recognize an individual by searching the entire database for a match. It conducts one-to-many comparisons to establish if the individual is present in the database and if so, returns the identifier of the reference that matched. In an identification system, the system establishes a subject's identity (or determines that the subject is not enrolled in the system database) without the subject having to claim an identity. Because identification in large databases is computationally expensive, a pre-selection stage is often used to filter the number of enrollment templates that have to be matched against the input feature set.

2.2 Image Capture and Fingerprint Sensing

In order to proceed with the processes of enrollment and verification/identification discussed, the issue of image capture must be considered. The two methods of capture are off-line and live-scan. Off-line capture is usually an inked impression on paper that then gets scanned. A live-scan image, on the other hand, is acquired by sensing the fingerprint directly by a scanner that digitizes the fingerprint on contact. For live-scan, typically a specialized fingerprint scanner is used. There are several live-scan capture methods to detect the ridges and valleys on the finger. Some of these methods include optical FTIR (Frustrated Total Internal Reflection), capacitive, thermal, pressure-based, and ultrasound. These are methods of sensing, and the sensor is the most important part of a fingerprint scanner. Almost all sensors are one of three types: optical, solid-state, and ultrasound.

Scanners can be classified into two groups: multi-finger and single-finger. Scanners also have the ability to capture *rolled impressions*, as opposed to regular flat impressions. Rolled impressions are taken “nail-to-nail” to provide an unwrapped version of a fingerprint compared to a flat print. While rolled impressions contain more detail, the rolling also introduces distortion that isn’t present on flat (plain) impressions. Hardware and software techniques have been introduced (see for example [11]) to enable live-scan fingerprint scanners to compose the sequence of images of the rolling finger into a single unwrapped impression.

In September 2010, our research group at Virginia Tech obtained an optical fingerprint scanner (*i3 digID mini*). It is capable of taking both flat prints and rolled prints. It uses the method of Frustrated Total Internal Reflection which will now be explained. The finger is pressed against the top side of a glass or plastic prism, so the ridges are in contact with the prism surface, and the valleys remain at a certain distance [see Figure 2.1]. A dispersed light source is directed through the left side of the prism, and the light is reflected at the valleys and randomly dispersed or absorbed at the ridges. The lack of reflection on the ridges makes them look dark on the produced image and they are thus distinguished from the valleys. The light rays exit from

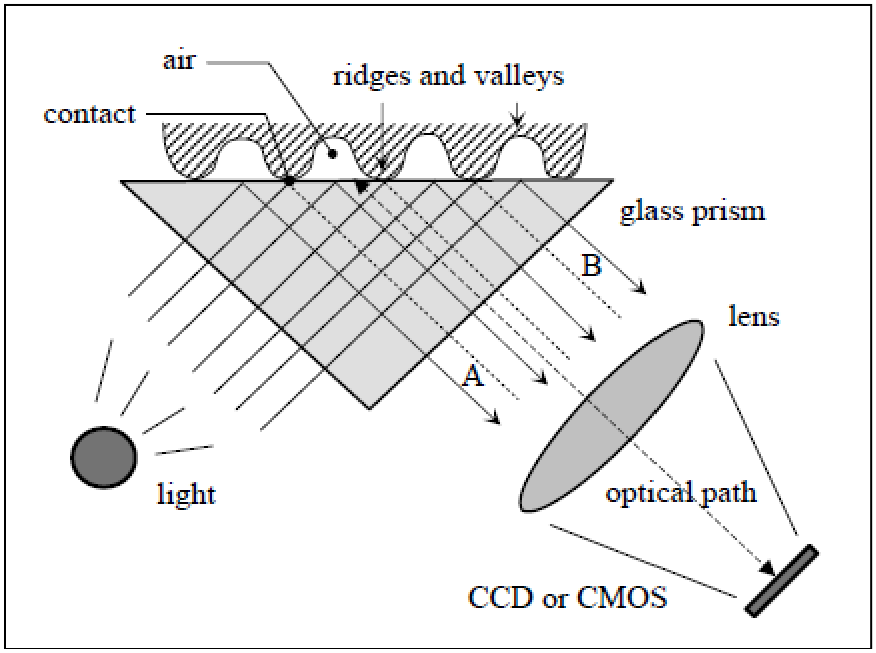


Figure 2.1. FTIR-based Fingerprint Sensor Operation
[from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

the right side of the prism and are focused through a lens onto a CCD or CMOS image sensor. Because FTIR devices sense a three-dimensional finger surface, they cannot be easily deceived by presentation of a photograph or printed image of a fingerprint. One issue that has to be taken into consideration when using this type of scanner is the moisture of the finger. When the finger is dry, it doesn't make uniform and consistent contact with the imaging surface. In this case, some ridges may appear too thin or partially missing. When the finger is overly moist, oils could also cover the valleys, which would cause the valleys in the image to shrink or disappear. Figure 2.2 shows the difference between images of a finger with varying moisture. Special wipes exist that one can apply to their fingers to give them the right amount of moisture if one's fingers are too dry.

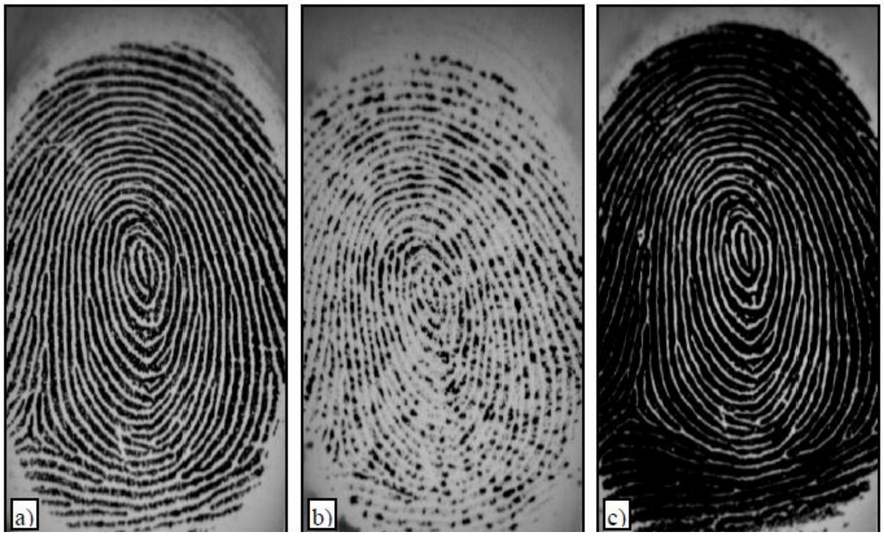


Figure 2.2. Three fingerprint images of the same finger with different skin conditions acquired with an optical FTIR scanner: a) normal, b) dry, c) wet
[from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

In order to maximize compatibility of matching fingerprints taken by different types of scanners, and to ensure high quality images, the FBI and other organizations involved in matching fingerprints released a set of standards and specifications for fingerprint scanners [12, 13]. It has been demonstrated that matching images of the same finger taken by different scanners, not compliant with given specifications, can greatly reduce fingerprint recognition accuracy [14-17].

For forensic applications, a particular kind of off-line image called latent fingerprints is of great interest. Latent fingerprints are those that are found at crime scenes and the like, where the oil from one’s skin leaves an impression of the ridges of the fingerprint on a material. The latent prints must then be “lifted” from the surface of the material by employing certain chemical techniques. Latent fingerprints are often poor quality and/or partial and thus present a challenge that must be overcome by fingerprint identification systems.

2.3 Capture of Latent Prints

As mentioned, latent fingerprints present a special challenge for AFIS, while there are also challenges to obtaining the latent print itself. Unlike tenprints obtained through live-scan or ink, the details of latent prints are largely invisible when deposited on the surface of a material. Thus they must be "lifted" and require methods to improve visibility and enhancements for identification. The prints are deposited due to the constant perspiration from sweat pores on the friction ridges which leaves a film of moisture on the surface of the contacted surface. Latent prints are often detected through the use of chemicals and optical techniques to capture the latent image revealed by the chemical. Powder dusting, ninhydrin spraying, iodine fuming, and silver nitrate soaking are the four most commonly used techniques of latent print development [2]. These techniques work well under normal circumstances but are not effective when the fingerprints are on certain surfaces (e.g., wet surfaces, untreated wood, human skin, etc.) [1]. Better procedures have been developed based on new chemical reagents, instruments, and systematic approaches involving a combination of methods (see [2, 18, 19]) to develop latent prints from such surfaces.

For the optical side of the process, a basic example is that laser-induced fluorescence from a fingerprint sample is photographed with a sensitive camera. From [19], "Optical filters are used to filter out the scattered excitation light as well as unwanted strong background emission from substrates. However, optical filtering often fails when the two emissions lie in a similar wavelength range, especially when the background emission is stronger in intensity than that of fingerprint emission."

2.4 Fingerprint Representation and Feature Extraction

The difficulty in fingerprint representation is finding such a representation that can discriminate between identities as well as remain invariant for a given individual. Thus the problem of representation is to determine a feature space in which the fingerprint images belonging to the same finger have similar characteristics in the feature space (low intra-class variations) and those belonging to different fingers occupy different portions of the space (high inter-class variations). A good fingerprint representation should have the following two properties: saliency and suitability [1]. Saliency means that a representation should contain distinctive information about the fingerprint. Suitability means that the representation can be easily extracted, stored in a compact fashion, and be useful for matching. Features that make a representation salient are not always suitable. Features that can be represented are discussed in the following sections.

2.5 Ridge Details

There exist three levels of ridge detail. Level 1 is the overall global ridge flow pattern. This consists of orientation of the ridges; focal areas (cores, deltas); and classification, which consists of type (arch, loop, and whorl) and ridge count. Level 2 is details of the ridge path, or simply the minutia points. Level 3 is specific ridge features, such as location of pores, local shape of ridge edges, and ridge width. Level 1 detail is only used for general classification of a fingerprint and is used in conjunction with Level 2 and 3 details for fingerprint matching. Level 2 features provide enough detail to show individualization with Level 1 detail, and Level 2 details (minutia points) are used extensively for fingerprint matching. Level 3 details are typically only persistent with high quality prints, and thus aren't very useful for latent print identification. However, in high quality prints level 3 details can be used for exclusion in matching.

Minutiae are typically denoted by their type (ridge ending or bifurcation), *x-y* coordinate location, and direction. The direction is defined as described in Figure 2.3.

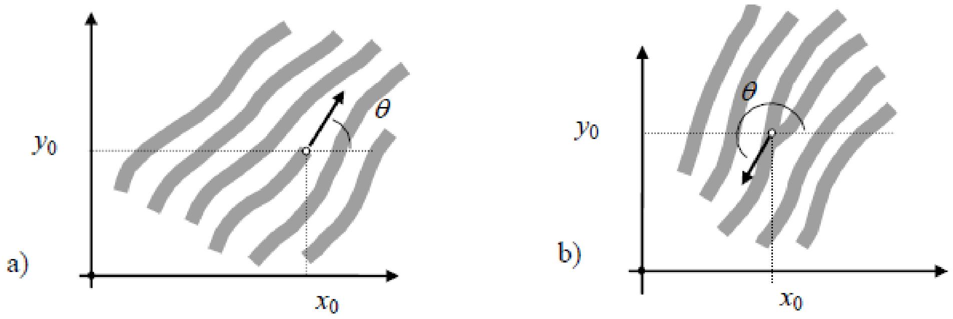


Figure 2.3. a) a ridge ending minutia: [x_0,y_0] are the minutia coordinates; θ is the angle that the minutia tangent forms with the horizontal axis; b) a bifurcation minutia: θ is now defined by means of the ridge ending minutia corresponding to the original bifurcation that exists in the negative image [from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

2.6 Ridge Count

Position, direction, and minutiae type aren't the only features that may be used for fingerprint recognition. Fingerprint experts and latent print examiners have often used ridge count to increase reliability in matching analysis. Ridge count is an abstract measurement of the distances between any two points in a fingerprint image [20]. Let **a** and **b** be two points in a fingerprint; then the ridge count between **a** and **b** is the number of ridges intersected by segment **ab** (Figure 2.4). Ridge counts have previously been used in some AFIS systems [21-23].

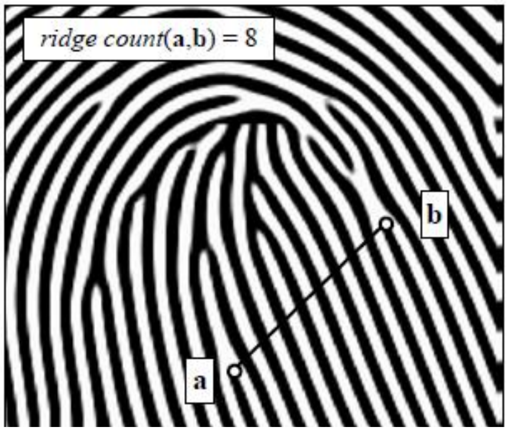


Figure 2.4. In this example the number of ridges intersected by segment **ab** (ridge count between **a** and **b**) is 8. [from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

2.7 Fingerprint Matching

Matching fingerprints is a difficult problem, mainly due to the amount of difference between two images of the same finger (intra-class variation). The main factors contributing to intra-class variation are rotation, displacement, partial overlap, non-linear distortion, pressure and skin condition, noise, and feature extraction errors. Also, fingerprints from different fingers may appear very similar when they are actually different, due to small intra-class variations and similar global structure. Typically, a match consisting of 12 minutiae points (the 12-point guideline) is considered as sufficient evidence in many courts of law [1]. Automatic fingerprint matching does not necessarily follow the same guidelines.

Automatic fingerprint matching algorithms often have little trouble matching high quality images; the difficulty lies in comparing low quality images as are often seen with latent prints.

Critical to a fingerprint matcher is a set of valid assumptions from which the design is based. Often there is a trade-off between using stricter constraints to ensure a good representation, and building a more sophisticated matcher for the given representation of the fingerprint. For instance, in a fingerprint matcher, one could constrain the elastic distortion and design the matcher based on an assumption of rigid transformation, or one could allow arbitrary distortions to accommodate the variations in the input images, for a more robust matcher. In light of this operational environment, the design of the matching algorithm needs to establish and characterize a realistic model or in some way take into account the expected variations among the representations of mated pairs.

2.8 Matching Module Errors

The result of a fingerprint matching module is typically a matching score that quantifies the similarity between the recognition feature set and the enrollment template. The higher the score the more certain the system is that the print comes from the same finger as the enrollment template. The decision module makes its decision by using a threshold *t*. Match scores higher than or equal to *t* are inferred as matching pairs (belonging to the same finger) and scores lower than *t* are inferred as non-matching pairs (belonging to different fingers).

When the matching module is operating in a one-to-one comparison mode (it compares a feature set from one finger with a template from one finger), it gives a match or non-match decision depending on whether the comparison score exceeds the threshold or not. The matching module can commit two types of errors: (1) mistaking the feature set and template from two different fingers to be from the same finger (called false match), and (2) mistaking the feature set and template from the same finger to be from two different fingers (called false non-match) [24].

It is important to understand the difference between false match and false non-match errors and the more commonly used false acceptance and false rejection errors. The false match and false non-match are errors of the matching module in one-to-one comparison mode while false acceptance and false rejection are the error rates associated with verification and identification processes. In fact their exact meaning is dependent upon the type of identity claim made by the user. For example, in applications with positive claim of identity (e.g., an access control system) a false match from the matching module results in the false acceptance of an impostor into the system, whereas a false non-match from the matching module causes the false rejection of a genuine user. On the other hand, in an application with negative claim of identity (e.g., preventing users from obtaining welfare benefits under false identities), a false match from the matching module results in rejecting a genuine request, whereas a false non-match from the matching module results in falsely accepting an impostor request.

Figure 2.5 shows the tradeoff between FNMR and FMR depending on where the threshold *t* is set. A system designer may not know in advance the application of the fingerprint system so it is typical to report performance at various operating points (threshold *t*). Figure 2.6 shows the Receiver Operating Characteristic (ROC) curve, graphically showing the FMR-FNMR tradeoff preferred by different applications. It illustrates that in forensic applications such as criminal identification, it is the false non-match rate that is of more concern than the false match rate; that is, we do not want to miss identifying a criminal even at the risk of manually examining[1] a large number of potential false matches identified by the system [1].

[1] For example, in criminal cases, an automated system produces a candidate set of possible matching prints that several examiners study in order to make a manual final determination.

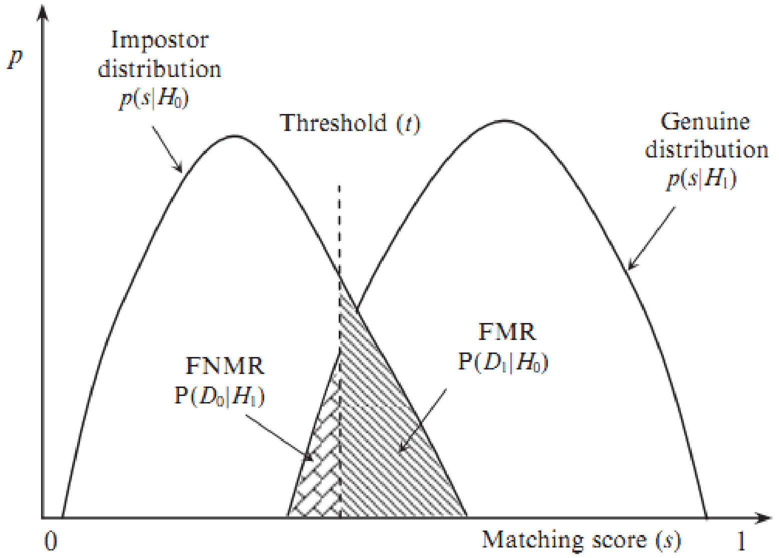


Figure 2.5. FMR and FNMR for a given threshold *t* are displayed over the genuine and impostor score distributions. Note that FMR is the percentage of genuine pairs whose comparison score is less than *t*. [from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

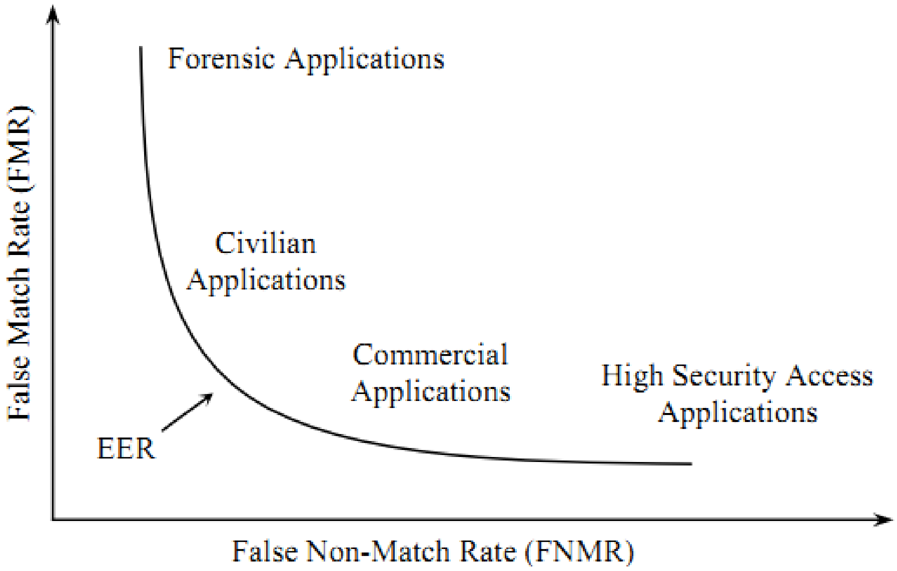


Figure 2.6. Typical operating points of different applications displayed on an ROC curve. [from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

2.9 Previous Work in Automated Fingerprint Matching

There are three main classes of fingerprint matching approaches. All three approaches have been attempted in automated systems to some degree. The three approaches are correlation-based matching, minutia-based matching, and non-minutia feature-based matching. Since the focus of this thesis is centered around minutia-based matching (along with ridge count and ridge information), the two non-minutiae based methods will be discussed first, and then we proceed to discuss minutia-based matching in more detail.

Correlation-based matching is basically a comparison on the pixels when two fingerprint images are superimposed. The correlation between corresponding pixels for different alignments (considering displacement and rotation) is calculated. The standard measure used, given a template $\boldsymbol{T}$ and an input $\boldsymbol{I}$, is the sum of the squared differences (SSD) between the intensities of the corresponding pixels:

$$SSD(\boldsymbol{T},\boldsymbol{I}) = \|\boldsymbol{T}-\boldsymbol{I}\|^2 = (\boldsymbol{T}-\boldsymbol{I})^T(\boldsymbol{T}-\boldsymbol{I}) = \|\boldsymbol{T}\|^2 + \|\boldsymbol{I}\|^2 - 2\boldsymbol{T}^T\boldsymbol{I} \tag{1}$$

where the superscript $\boldsymbol{T}$ denotes the transpose of a vector. If the terms $\|\boldsymbol{T}\|^2$ and $\|\boldsymbol{I}\|^2$ are constant, the diversity between the two images is minimized when the cross-correlation (CC) between $\boldsymbol{T}$ and $\boldsymbol{I}$ is maximized:

$$CC(\boldsymbol{T},\boldsymbol{I}) = \boldsymbol{T}^T\boldsymbol{I} \tag{2}$$

Due to displacement and rotation between two impressions of the same finger, the similarity cannot be computed just from superimposing $\boldsymbol{T}$ and $\boldsymbol{I}$ and applying equation (2). The problem with correlation-based matching is that non-linear distortion makes impressions of the same finger significantly different in terms of global structure. While elastic distortion does not have much effect locally, the effects get integrated in the image space so two global fingerprint patterns cannot be reliably correlated. Also, skin moisture and finger pressure cause image brightness, contrast, and ridge thickness to vary significantly across different impressions. Other more sophisticated correlation measures are necessary.

Non-minutia-based feature matching techniques are often used in addition to minutia-based techniques to increase system accuracy and robustness. We consider these techniques independently from minutia-matching techniques, however. In the case when the area of the fingerprint is small, only 4-5 minutiae might exist, and in that case minutiae-based matching algorithms don't work satisfactorily, so non-minutia-based methods may perform better. The more commonly used non-minutiae features are:

1. Size of the fingerprint and shape of the external fingerprint silhouette.
2. Number, type, and position of core and delta points.
3. Global and local texture information.
4. Geometrical attributes and spatial relationship of the ridge lines.
5. Level 3 features (e.g., sweat pores).
6. Other features: fractal features [25], shape features derived from the one-dimensional projection of the two dimensional fingerprint image [26-28].

Features 1 and 2 are unstable. Features 3, 4, and 5 are very useful for automatic fingerprint matching.

Global and local texture information: The most popular technique to match fingerprints based on texture information is the FingerCode approach by Jain et al. [29] In this method an area of interest is tessellated into cells, and a feature vector is composed of an ordered enumeration of the features extracted from a cell. FingerCodes are not as distinctive as minutiae, but carry complementary information that can improve overall matching accuracy.

Geometrical attributes and spatial relationship of the ridge lines: This method involves sampling points at fixed intervals along ridges. Ridge information can be combined with minutia information to account for the spatial matching of the ridge points. Ridge information can also be explicitly used and mapped into feature vectors. Two ridges can be considered a match if they have similar length and curvature. Examples of explicitly exploiting ridge relationships can be found in [30-32]. Xie et al. match ridges from the ridge skeletons if they have similar length and curvature [30]. Feng et al. begins with a local alignment based on a minutia and its adjacent ridges [31]. Then ridge-minutia pairs are incrementally matched and a match score is computed. Feng and Cai use a Ridge Coordinate System (RCS) to encode the geometric relationships between every minutia and sampled points on all other ridges to align and then match using a greedy algorithm [32].

Level 3 features: Reliable detection of sweat pores requires high-resolution scanners (at least 800 dpi) and robust extraction algorithms. If pore information can be reliably extracted and matched, it has been shown that the probability of occurrence of a particular combination of 20 ridge-independent pores[2] is 5.186×10^{-8} [33].

Now we discuss minutia-based matching. Unlike correlation-based techniques where the fingerprint representation coincides with the fingerprint image, here the representation is a feature set where the features are minutiae. Most minutiae matching algorithms consider each minutia as a triplet $\mathbf{m} = [x,y,\theta]$ that indicates the x,y minutia location coordinates and the minutia angle θ. The minutia points between the template **T** and the input **I** are compared.

$$\begin{array}{lll} \mathbf{T} = \{\mathbf{m}_1, \mathbf{m}_2, \ldots \mathbf{m}_m\}, & \mathbf{m}_i = \{x_i, y_i, \theta_i\}, & i = 1\ldots m \\ \mathbf{I} = \{\mathbf{m}'_1, \mathbf{m}'_2, \ldots \mathbf{m}'_n\}, & \mathbf{m}'_j = \{x'_j, y'_j, \theta'_j\}, & j = 1\ldots n, \end{array}$$

The distance between the minutia points (spatial difference, or *sd*) being compared and the difference in their angles (direction difference, *dd*) need to be taken into account. Both distance and angle differences need to be below a given threshold to be considered a match.

[2] The term “ridge-independent” is used because the authors of [28] state that it may be unnecessary or too difficult to associate pores with specific ridges, so disregarding their ridge association is preferable.

$$sd(\boldsymbol{m'}_j, \boldsymbol{m}_i) = \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \leq r_0 \tag{3}$$

$$dd(\boldsymbol{m'}_j, \boldsymbol{m}_i) = min(|\theta'_j - \theta_i|, 360° - |\theta'_j - \theta_i|) \leq \theta_0 \tag{4}$$

The tolerances r_0 and θ_0 are needed to compensate for feature extraction errors and the small elastic distortions that cause the minutia positions to change. To do the comparison, typically the fingerprints need to be aligned to account for displacement and rotation, in order to produce the most matches between corresponding minutia points. Scale should be considered in the case that the resolution of the prints varies, and distortion-tolerant geometric transformations could be applied in case the fingerprint is affected by severe distortion. In the case where the correct alignment isn’t known, a mapping function could be used to match each minutia in the template with its closest match in the input print. This is more of an iterative approach and would be more expensive in terms of time.

There have been other proposals to avoid alignment. These approaches would fall under either the global minutia matching or local minutia matching categories. One approach to match minutiae globally without alignment involves an intrinsic coordinate system, based on the ridge flow of the print. Local minutiae matching consists of comparing two fingerprints according to local minutiae structures. Local structures by definition aren’t affected by the global transformations, translation and rotation. However, by eliminating the global spatial relationships which are highly distinctive, we reduce the amount of information available for discriminating fingerprints.

In the case of manual matching performed by forensic experts, the means to determine a match is the number of matching minutiae. For automated matching systems, this number must be converted to a similarity score. This is often performed by simply normalizing the number of matching minutiae (denoted by *k*) by the average number of minutiae in the template **T** and input **I** (shown as *n* and *m* respectively).

$$score = \frac{k}{(n+m)/2}$$

However, the similarity score can be improved by including weights for minutia quality/reliability.

Minutiae matching can be generalized to the well-studied *point pattern matching* problem, used in pattern recognition and computer vision tasks.

Various algorithms have proposed various local structures used for local matching. Local matching is invariant to distances and angles between the images being compared. In [21] local structures are formed by a central minutia and its two nearest-neighbor minutiae. The feature vector of this structure for a minutia $\mathbf{m}_\mathbf{i}$ is:

$$\boldsymbol{v}_i = [d_{ij}, d_{ik}, \theta_{ij}, \theta_{ik}, \varphi_{ij}, \varphi_{ik}, n_{ij}, n_{ik}, t_i, t_j, t_k],$$

where d_{ab} is the distance between minutiae $\mathbf{m}_a$ and $\mathbf{m}_b$, θ_{ab} is the direction difference between the angles θ_a and θ_b of $\mathbf{m}_a$ and $\mathbf{m}_b$, φ_{ab} is the direction difference between the angle θ_a of $\mathbf{m}_a$ and the direction of the edge connecting $\mathbf{m}_a$ and $\mathbf{m}_b$, n_{ab} is the ridge count between $\mathbf{m}_a$ and $\mathbf{m}_b$, and t_a is the minutia type of $\mathbf{m}_a$ (Figure 2.7).

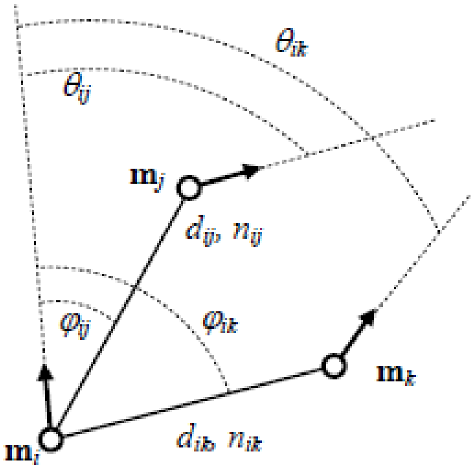


Figure 2.7. Features of the local structures used by Jiang and Yau. [from Maltoni et al., *Handbook of Fingerprint Recognition (2nd Ed.)* (2009)] [1]

The matching of local structures invariant with respect to translation and rotation, forms the basis for most of the minutiae matching algorithms after 2000 [1].

In automated fingerprint matching, the idea of using triplets to group minutiae for matching purposes is not new. A localized triangular feature provides distortion tolerance and computational efficiency when considering relations between the points of the triplet. Minutiae triplets have been used for indexing, to increase search speed and reduce potential match candidates, as in [34, 35]. Minutiae triplets also have been used for straight minutiae-matching algorithms, as in [36, 37]. The works of [34, 35, 37] particularly emphasize the usefulness of their triplet or triangle-based algorithms when dealing with the effects of non-linear distortion. The work of [38] is similar to our work in that it uses the triangle as a base local feature and extends modularly to include other triangles in a global scope. Various features of triangles have been used as distinguishing features: the length of each side; the angles of the ridges; the ridge count between minutiae; triangle handedness; triangle type; triangle direction; maximum side; minimum and median angle; minutiae density in a local area; ridge counts along the three sides; for each vertex, the average deviation of the local orientations in a neighborhood. Triangles have also been used as an initial aligning feature, and then other minutiae are matched based on the given alignment.

Until now only general methods and techniques have been described. Now one particular fingerprint matching algorithm is described. The algorithm is BOZORTH3, which was included in the most recent release of NIST fingerprint software [39]. BOZORTH3 computes a match score between the minutiae from any two fingerprints to help determine if they are from the same finger. It uses only the location (x,y) and orientation (θ, or t) of the minutia points to match the fingerprints. The matcher is rotation and translation invariant. Now the algorithm will be described. First, intra-fingerprint minutia comparison tables are constructed. For this step, for each fingerprint, entries are created in the table for each pair of minutiae in the print. The data stored in the table consists of: [d_{kj}, β_1, β_2, k, j, θ_{kj}], where d_{kj} is the distance between the two points being compared; angles $\beta 1$ and $\beta 2$, where $\beta 1$ is the angle between the theta orientation of minutia k and the line connecting the two minutiae, and $\beta 2$ is the same for minutia j; the positions of the two minutia points k and j; and an angle θ_{kj}, the arctangent of the slope of the connecting line (see Figure 2.8). In Figure 2.8, P_m indicates a minutia from the probe print and G_n indicates a minutia from the gallery print.

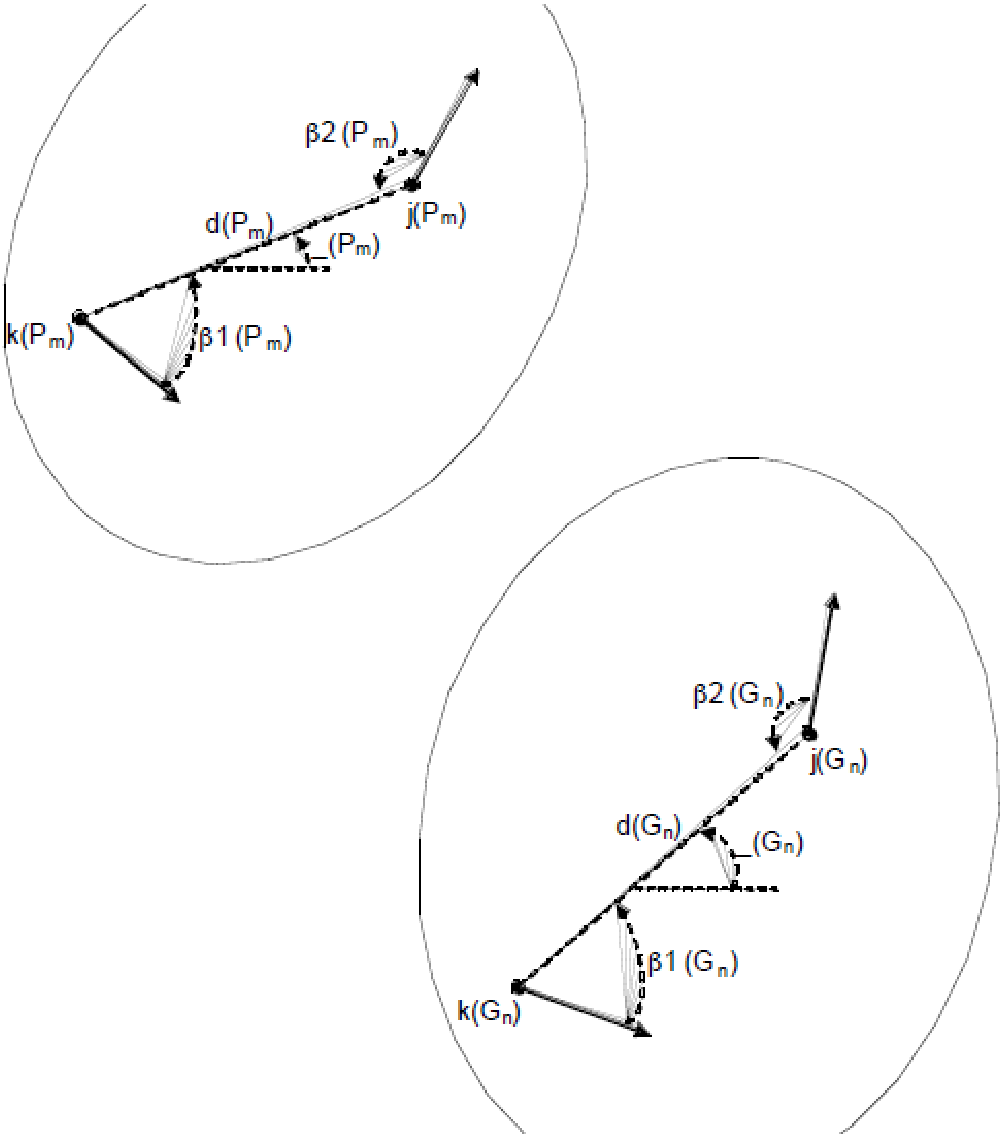


Figure 2.8. Intra-fingerprint minutiae comparison. [From “User's Guide to Export Controlled Distribution of NIST Biometric Image Software” (2004)]

The next step is to construct the inter-fingerprint compatibility table. This involves taking the comparison tables from two separate fingerprints and looking for compatible entries between the two tables. To determine compatible entries, three criteria are used. The distance value must be within a certain threshold, and the two angles β1 and β2 must be within a certain angle threshold. If a minutia pair passes the compatibility thresholds, it is entered into the compatibility table. An entry in the compatibility table looks like this: $[\Delta_{\beta}(\theta(P_m),\theta(G_n)), k(P_m), j(P_m), k(G_n), j(G_n)]$. This association represents a single link in a compatibility graph. Then, to determine how well the two fingerprints match, you would traverse the links to find the longest path. The match score would be the length of the longest path. There are issues to this approach, however, that the algorithm accounts for. Some of these issues are that the graph is really a collection of disjoint links, each node could have multiple links, there could be loops, there is no obvious root that would produce the longest path, etc. So there exists an algorithm that traverses the compatibility table to form some type of compatibility clusters. The number of linked table entries across the combined clusters is accumulated to form the actual match score. The match score is an indicator of how likely that the fingerprints are a match.

Another algorithm that was presented in 2002 will be discussed [40]. This algorithm uses not only position, angle, and type of minutia data, but also associated ridge data. In this algorithm an initial alignment is used, based on a template ridge, to align the prints at the same angle. Every minutiae angle gets adjusted by the amount of the alignment angle. Each minutia point in the template and input prints gets converted to polar coordinates with respect to a reference minutia in each print. The minutiae are represented as symbolic strings in the polar coordinate system, where each minutia is concatenated to the string based on increasing radial angle. The strings of P_i and Q_j get a matching score which is recorded as m_score[i][j]. After all P_i and Q_j pairs have been evaluated, take the highest score from m_score[i][j] as the matching score of the template and input minutiae set. If this is greater than some threshold value then the pair is identified as a match. A non-linear bounding box is used to handle nonlinear deformation more robustly.

Many other algorithms exist, although most commercial algorithms are not published. Competitions such as FVC2000 and FVC2002 aimed to determine the state-of-the-art in fingerprint matching and perform a technology comparison. In 2007 NIST conducted a series of tests to determine the state-of-the-art in latent fingerprint matching, called “Evaluation of Latent Fingerprint Technologies 2007” (ELFT07). Participants included vendors of Automated Fingerprint Identification Systems (AFIS). Effects were studied such as image resolution, using a Region of Interest (ROI), the number of minutiae present in a latent print, which finger gives the best results, and which type of print gives the best results.

2.10 Background on “Sufficiency”

In the NIJ solicitation “Forensic Friction Ridge (Fingerprint) Examination Validation Studies” [10], the NIJ expressed that they were looking for “analyses on an objective and scientific approach for determining the sufficient quality and quantity of friction ridge details needed to conduct a valid examination for comparison purposes.” From [8], referring to the solicitation by the NIJ: “The two main topics of basic research under this solicitation included: 1) measure the amount of detail in a single fingerprint that is available for comparison and 2) measure the amount of detail in correspondence between two fingerprints.” Together these are three notions to help explain the meaning of *sufficiency* as we use it. There is also a fourth notion. Explained in my own words, the four notions of sufficiency that we attempt to advance are: 1) determining if a latent print contains sufficient amount and quality of detail to proceed with an examination; 2) determining if that detail is sufficient in amount and quality to proceed with making any comparisons; 3) determining if that detail is sufficient in amount and quality to identify a match between two fingerprints; 4) determining if that detail is sufficient to claim that the fingerprint must belong to a certain individual and that individual only (identification).

2.11 Previous work on individuality models

Another area of previous work related to this thesis is the development of individuality models. Pankanti et al. [8] offer estimates of probabilities of false correspondence, while developing other conclusions such as that automated fingerprint matchers' performance is not close to the theoretical limit. Chen and Moon [9] develop their own individuality model based on an assumption of uniform distribution of minutiae, and then in later work state that the distribution is not uniform and develop a stochastic model [41]. Zhu et al. developed models that better represent the clustering property of minutiae features and they compute the probability of random correspondence [42]. A more thorough review of individuality models can be found in [1].

Perhaps one of the most similar works relating to this one is by Neumann et al. [43], which proposes the use of likelihood ratio to assess the value of fingerprint evidence and uses minutia triplets to do so. However, in this work, we instead develop our concept of finding distinctive features that occur rarely in any given population of fingerprints, and augment our minutia triplets with the shared-ridge segment idea and ridge counts.

Chapter 3: **Evaluating Sufficiency Using Minutia Triangles with Ridge Information in a Matching Algorithm**

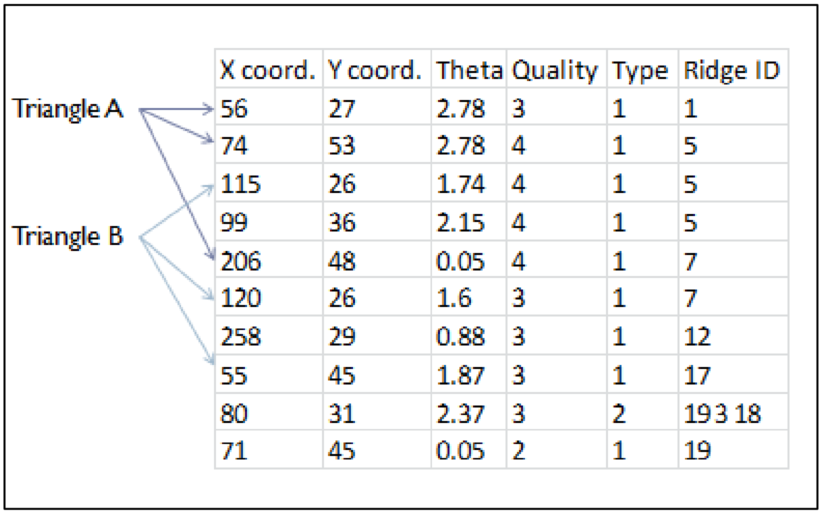
3.1 Introduction to Minutia Triangle Features

Empirical evidence suggests that certain distributions of minutiae or fingerprint features have a strong influence on matching, quality assessment, and sufficiency. Most of the existing techniques focus on minutiae only. Our work includes not only minutia-based assessment, but incorporates ridge-based relations between minutiae or groups of minutiae. We believe that by relating minutiae with surrounding friction ridges, much more information can be gathered for the purpose of this work.

As a prelude to considering more general feature distributions, we establish a groundwork involving 3-point sets of minutiae, each of which can be easily visualized as triangles in the fingerprint image. We refer to the terms “triplets” and “triangles” interchangeably. The approach initially bears similarity to work described by [35]. A major difference, however, is our incorporation of information from ridges, particularly the ridge direction at each minutia. While other researchers have included ridge information in a minutia-based matching scheme [31, 44-46], they did not use ridge information as we propose; its value shall become clear in later discussion. Further, our algorithm was designed with computational efficiency in mind, making comparison on a large fingerprint database practical [47]. Thus, we have identified trade-offs that can be made to strike the proper balance of accuracy vs. computational load. For example, we avoid the relatively expensive trigonometric and divide calculations used by [34, 36, 48-51] by ignoring the geometric angles of the triangles.

3.2 Triplet-Based Descriptor/Framework Overview

First, we must have a listing of all extracted minutiae from a fingerprint, which serves as the input to our analysis framework. Many existing tools can generate such files from a fingerprint image. The input is in the form of an “xyt” file, where *x* and *y* are the *x-y* coordinates of the minutia and *t* is theta (θ), or the angular direction of the associated ridge of the minutia (ridge orientation). The “xyt” file is also augmented with ridge ID numbers for each minutia with our internally developed engine (by Nathan Short, Virginia Tech). This ID number indicates which ridge(s) a minutia is part of. Using these extended “xyt” files, any combination of three minutiae forms a triplet, or triangle (see Figure 3.1).


Triangle A
Triangle B

X coord.	Y coord.	Theta	Quality	Type	Ridge ID
56	27	2.78	3	1	1
74	53	2.78	4	1	5
115	26	1.74	4	1	5
99	36	2.15	4	1	5
206	48	0.05	4	1	7
120	26	1.6	3	1	7
258	29	0.88	3	1	12
55	45	1.87	3	1	17
80	31	2.37	3	2	19 3 18
71	45	0.05	2	1	19

Figure 3.1. Sample portion of an “xyt” file

These triangles are each supplemented with attributes to form a triangle-based descriptor. The attributes used initially are: (1) side lengths of the triangle (pairwise distances between minutiae), (2) differences between pairwise ridge orientations (Δθ values), and (3) type of each minutia in the triangle. Additionally, we consider (4) ridge count, or ridges intersected, between pairwise minutiae, and (5) whether each pairwise minutiae lies on a shared-ridge segment. The ridge counts are obtained through lookup in a database of ridge counts for all pairwise minutiae for all fingerprints. However, we note that all or only a subset of these attributes may be used; for a given matching situation, we can benefit from using a case-based approach to best utilize available data. For example, it may not be feasible to use minutia type as a distinguisher, since minutia type is known to be difficult to accurately determine during the minutiae extraction process [52, 53] and may therefore be unreliable in lower-quality regions of the image. For the purpose of this work, we assume all features extracted (minutia type, ridge count, etc. to be reliable. The basic triangle descriptor can be represented by the image shown in Figure 3.2, with ridge information shown in Figure 3.3.

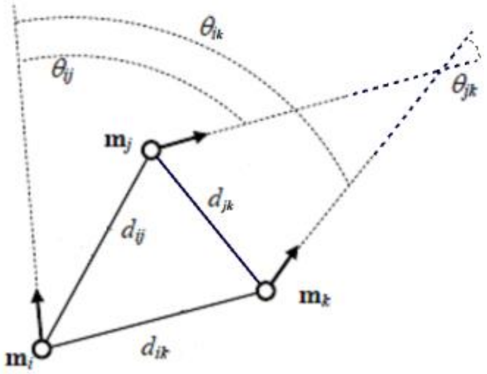


Figure 3.2. Triangle formed by 3 minutiae

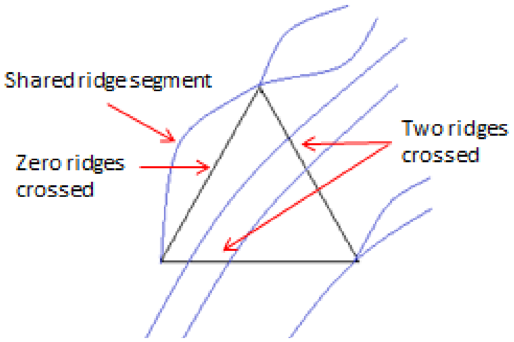


Figure 3.3. Extended ridge-based triangle attributes

One of the main benefits to our triangle-based method is that it is rotation- and displacement-invariant (illustrated in Figure 3.4), which can be powerful in making assessments about the image. The benefit to being rotation- and displacement-invariant is that any relevant features identified can apply to an image of any orientation or translation. This may play a role when analyzing a distinctive feature identified on a latent print. For example, the latent may be shifted to one side, or some of the database prints being compared may be rotated. Rotation- and displacement-invariance is achieved by focusing on distances (relative spacing and ridge counts

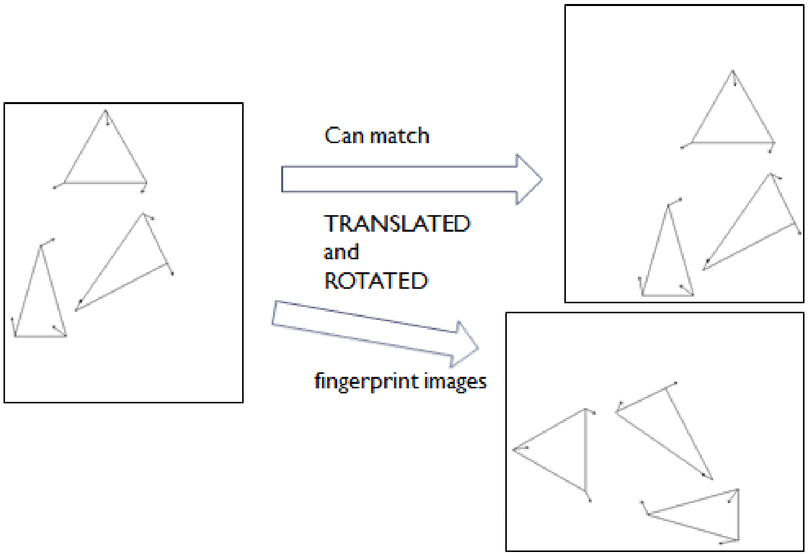


Figure 3.4. Illustrating matching against translated or rotated images

between minutia points) and angle differences. We also choose to restrict the maximum side length of a triangle to keep the triangle as a localized feature, thus eliminating possible distortion that occurs over long distances.

3.3 Extension to multi-triangle feature

As any combination of attributes on a single pair of matching triangles may not provide sufficient information to ensure that two fingerprint images could highly likely have originated from the same finger, we extend the single triangle to 2 or 3 triangles (up to 6 or 9 minutiae). By extending the feature size to 6 or 9 points while still using the triangle as a base, we achieve a *modular* and *hierarchical* framework that enhances discriminatory power without much greater computational cost. The approach is similar to that of [38].

When we consider multi-triangle features, we must associate more attributes to such a descriptor to distinguish such cases as shown in Figure 3.5. In Figure 3.5, triangle 1 in the reference (left) image matches triangle 1 in the right image. Similarly, the triangles labeled ‘2’ were marked as a match, since matching is rotation invariant. However, when we consider the

"2-triangle (6-point) feature" composed of triangles 1 and 2, the features can easily be seen as non-matching.

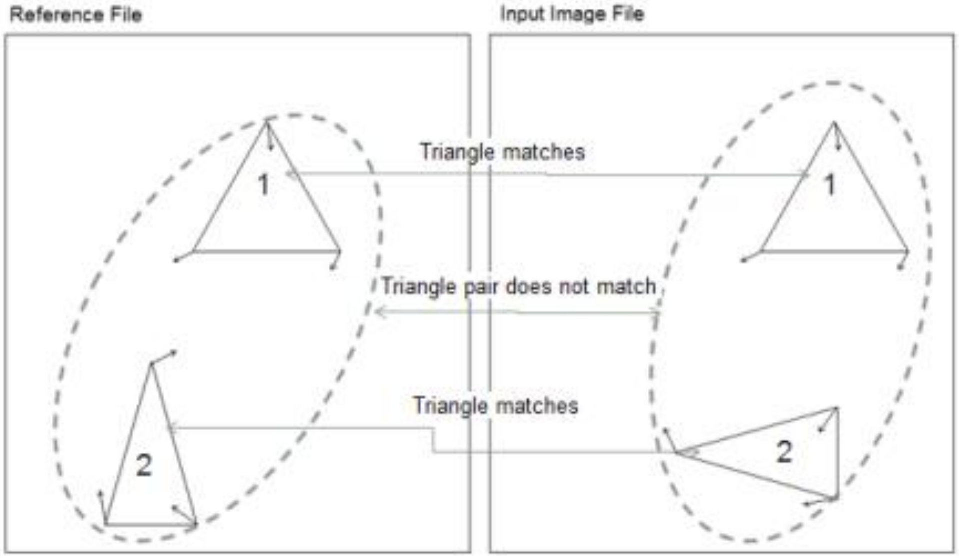


Figure 3.5. Comparison of two-triangle sets

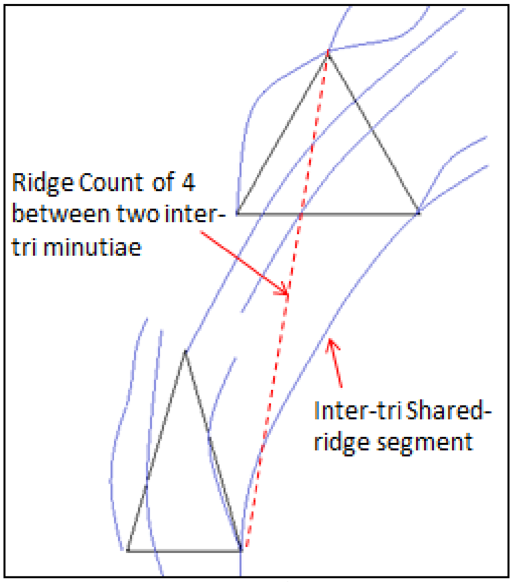

Ridge Count of 4 between two inter-tri minutiae
Inter-tri Shared-ridge segment

Figure 3.6. 2-triangle feature with inter-triangle shared ridge segment and inter-triangle ridge count indicated

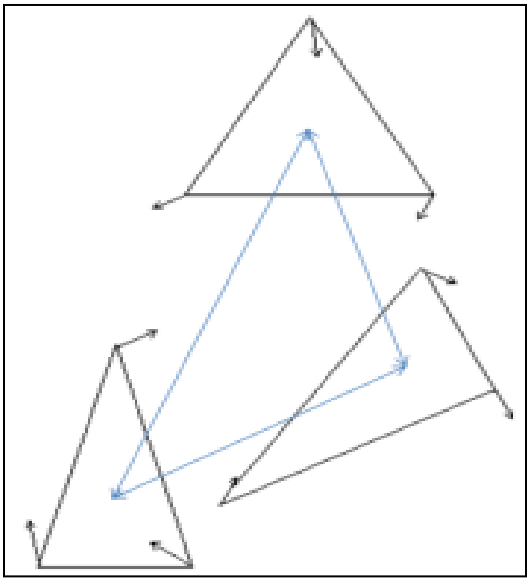


Figure 3.7. Illustrating the distance between centroids as a distinguishing trait

A new set of attributes to consider opens up when considering multi-triangle features: those of inter-triangle relations. Figure 3.6 illustrates the ridge relations we consider between two triangles. Specifically, we consider the ridge count between pairwise minutiae between two different triangles (inter-triangle ridge count), and also look for shared-ridge segments between minutiae of different triangles (inter-triangle shared-ridge segments). (In Figure 3.6, a ridge count of 4 is shown because our extraction algorithm includes the ridges at the minutia intersections.) Such information could be immensely useful and help add value to the information embedded in the image. We also consider the distance between the centroids of the triangles (Figure 3.7) and the differences in pairwise $\Delta\theta$ values between triangles.

As we analyze a latent image containing 6 or more minutiae, such multi-triangle features will provide immense value when comparing against a set of ten-print images of any suspects. One could first check if any ten-print contains similar multi-triangle features. However, the confidence in the match may not be too high if the multi-triangle feature can be found in many other fingerprints. Stated differently, the confidence would be increased if the matched multi-triangle feature from the latent is rare, that is, the feature occurs very infrequently in the population (see Chapter 4).

Clearly, our approach of using 2 or more triangles, plus attributes that enhance discrimination power, allows us to have a modular and hierarchical algorithm that can boost accuracy, without much effect on computational cost.

3.4 Triangle Feature-based Matching Algorithm

Our goals here are to identify how much information in a given fingerprint would be sufficient to identify a “match” candidate to another fingerprint of the same finger, and to identify sufficient information to ensure that no other images in the set would be a match candidate (identify a unique match). To do this we developed a minutiae triangle-based matching algorithm.

First, we adopted a simplistic matching algorithm that considered only the minutia-based information of x, y, and θ for each minutia. Then we compared these results with those incorporating other ridge information. From the input “xyt” file discussed previously, all enumerations of triangles are considered that fall below the maximum side length restriction and that fall above the minimum side length restriction (used to minimize the impact of falsely extracted minutiae that lie close together). Refer to Figure 3.2 for a depiction of the extracted triangles. The initial matching criteria are:

- Sum of side lengths and length of shortest side
- Sum of $\Delta\theta$ values and smallest $\Delta\theta$ value
- Type (ending or bifurcation) of each minutia in the triangle

The $\Delta\theta$ value mentioned above is the difference in θ values between two minutia points or vertices of the triangle. Hence each triangle has three $\Delta\theta$ values. As previously stated, our algorithm is modular and we may choose whether to include the type identifier as a match criterion as well as use of the ridge information.

Our algorithm also should not be too computationally expensive so as to make a comparison on a large fingerprint database infeasible [47], so in some cases trade-offs have been made, sacrificing some discriminatory power for reduced computational load. For example, we avoid the relatively expensive trigonometric and divide calculations used by [34, 36, 48-51] by ignoring the geometric angles of the triangles and other geometric considerations. Also, *squared distances* are used in place of the usual Cartesian distance formula in order to avoid the expensive square root computation.

The algorithm also will tolerate a certain amount of distortion to accommodate the often low-quality images of latent fingerprints, just as Parziale admits a certain amount of deformation

[36]. This is achieved through tolerance parameters that identify matches within a certain similarity threshold.

Tolerance values are required to determine if two given features are “similar enough” to be deemed a match. There are two types of tolerance values required. The first type of tolerance value is the distance/spatial difference tolerance. The spatial difference tolerance parameter (abbreviated as *sd*) is expressed as a percent difference. Thus, a *sd* tolerance parameter of 5 corresponds to an allowable 5% difference tolerance in compared spatial differences. Expressing the *sd* tolerance as a percent difference has the advantage that it takes into account the increased translational distortion possible the farther the two points are away from each other.

The second type of tolerance is the angle tolerance, which is used to determine if two compared $\Delta\theta$ direction differences are similar enough to be matched. The angle tolerance is always used when two *differences* between angles (directions/ridge orientations) are being compared. Here, the tolerance cannot be expressed as a percent difference, since for example, the difference between 1 degree and 2 degrees is very small, however that is a 100% difference, whereas the difference between 99 and 100 degrees is the same but the percent difference is very small. Thus we need an absolute tolerance scale, so we express the tolerance as a number of degrees. This is illustrated in terms of the unit circle in Figure 3.8. We refer to this tolerance value as the “base” $\Delta\theta$ tolerance value, which is used to compare one $\Delta\theta$ (direction difference) value to another. When we compute the sum of several $\Delta\theta$ and compare that to another sum of $\Delta\theta$, the potential distortion from each individual $\Delta\theta$ could potentially add up. Thus we choose a conservative tolerance value equivalent to a multiple of the base $\Delta\theta$ tolerance value, depending on how many $\Delta\theta$ values are being summed.

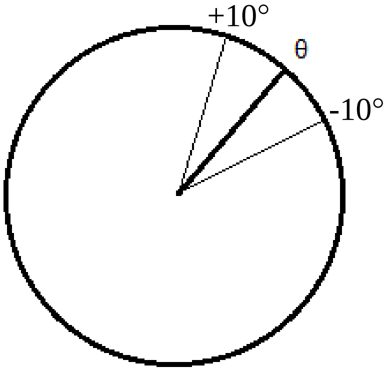


Figure 3.8. Illustrating the allowable distortion tolerance for direction difference comparisons

Another performance consideration is that comparisons based on $\Delta\theta$ values and type are always made before distance (spatial difference) comparisons. This is because the computation for distance is more expensive, so we avoid making the computation if $\Delta\theta$ mismatch or type mismatch already rule two triangles as non-matching.

As stated, we have used an exhaustive approach to evaluating 3-point minutia sets, or triangles. For a given input image, we identify every possible triangle (meeting minimum and maximum side length restrictions) using the extracted minutiae. Then each triangle from the input image is compared with every possible triangle from all images in a database (illustrated in Figure 3.9). The algorithm decides that 2 triangles match if all of the attributes differ by no more than the allowable tolerance, such as 5%.

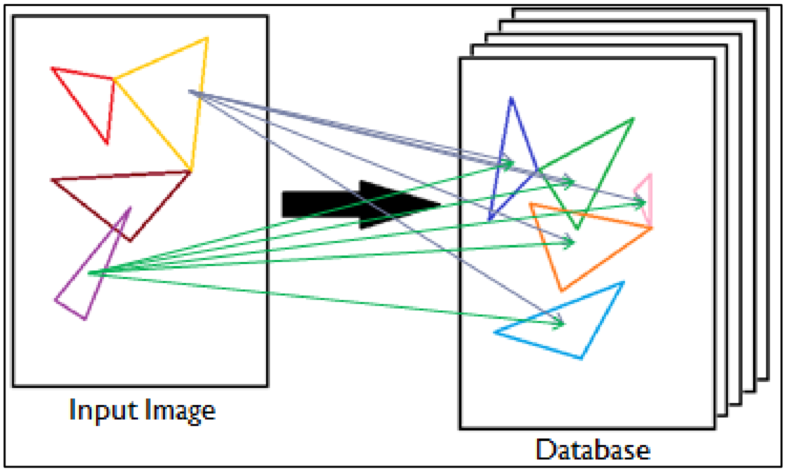


Figure 3.9. Illustrating exhaustive approach to evaluating triangle comparisons on the database

However, a pair of matching triangles is not nearly sufficient to say that two fingerprint images match, as many false matches can be produced. So the matching feature must then be extended to increase discriminatory power. As mentioned, we then consider 2- or 3-triangle feature sets (see Figures 3.5-3.7).

3.5 Possible Matching Tests

In Table 3.1 we provide a list of all possible matching tests developed, followed by descriptions of the multi-triangle tests and the single triangle tests not already mentioned.

Table 3.1. List of Possible Matching Tests

Single Triangle Tests:	**Multi-Triangle Tests:**
Sum of Δ*θ*s, smallest Δ*θ*	No overlap restriction
Sum of side lengths, smallest side length	Inter-triangle shared ridge count
Type count	Shared ridge required restriction
Both minutia types required restriction	Type match of inter-triangle shared ridge minutiae
Shared ridge count	Inter-triangle shared ridge minutiae Δ*θ*
Shared ridge minutiae Δ*θ*	Distances between inter-triangle shared ridge minutiae
Distance between shared ridge minutiae	Distances between centroids
Type match of shared ridge minutiae	Distances between pairwise inter-triangle minutiae
Ridge count	Inter-triangle distances between unique type minutiae
	Sum and smallest inter-triangle Δ*θ*s for different types
	Sum and smallest inter-triangle Δ*θ*s
	Ridge count

Single Triangle Tests -- Shared Ridge Minutiae $\Delta\theta$ Test, Distance Between Shared Ridge Minutiae Test, and Type Match of Shared Ridge Minutiae:

These tests are additional properties that should hold if two triangles from two different impressions are indeed from the same finger (a true match). If the candidate matched triangles were identified as having a shared ridge segment, and if the finger source is the same, then it should also hold that the $\Delta\theta$ values between the minutiae attached to the shared ridge should match, the distance between those minutiae should match, and the types of the minutiae on both triangles should be the same. Again, the attributes "match" if they are similar within the thresholds of the given distortion tolerance parameters.

Multi-Triangle Test – No overlap restriction:

In the algorithm, once we determine all matching triangle pairs, all enumerations of triangles are formed into 2-triangle or 3-triangle sets. In this case there exist many cases where one or two vertices are shared by two triangles. For example, in the case of looking at 2-triangle features if we allow overlap to occur, the two triangles may constitute 4, 5, or 6 minutiae depending on if minutiae are overlapped between the two triangles. By insisting that there exist no overlap, we confine 2-triangle and 3-triangle features to 6 and 9 points, respectively.

Multi-Triangle Test – Inter-Triangle Shared Ridge Count:

We look for shared ridges between triangles (inter-triangle shared ridges) and make sure that 2- or 3-triangle features being compared have the same number of them.

Multi-Triangle Test -- Shared Ridge Required Restriction:

As a shared ridge segment provides very useful information we can use to assess quality, we may restrict the matches we look for to only those multi-triangle features that contain a shared ridge segment between two of its minutiae.

Multi-Triangle Tests – Inter-triangle Shared Ridge Minutiae Δθ Test, Distance Between Inter-triangle Shared Ridge Minutiae Test, and Type Match of Inter-Triangle Shared Ridge Minutiae:

See the single triangle tests of similar name above. The same applies for existing inter-triangle shared ridges – the involved minutiae should have matching Δθ, matching distances between them, and matching types.

Multi-Triangle Tests – Distances Between Centroids:

We must ensure that the spatial differences between the multi-triangle feature sets match. One way to do this is to use the distance between the centroids of the triangles to do this. In a 2-triangle feature, only one centroid-to-centroid distance exists. However, in a 3-triangle feature, there exists three centroid-to-centroid distances (see Figure 3.5), forming a centroid triangle. We compare these centroid distances as we do the side lengths of a triangle – checking that the sum of the distances match as well as the smallest distance.

Multi-Triangle Tests –Distances Between Pairwise Inter-Triangle Minutiae:

Another way to consider the spatial distances between the triangles is to consider the distances between all inter-triangle minutiae. The sum of all pairwise distances is compared. This test or the previous test can be used, but it would be inefficient to use both since they both are a check of the same thing – spatial differences between triangles. (A head-to-head comparison of both these methods will be shown later to determine which is better.)

Multi-Triangle Test -- Inter-Triangle Distances Between Unique Type Minutiae:

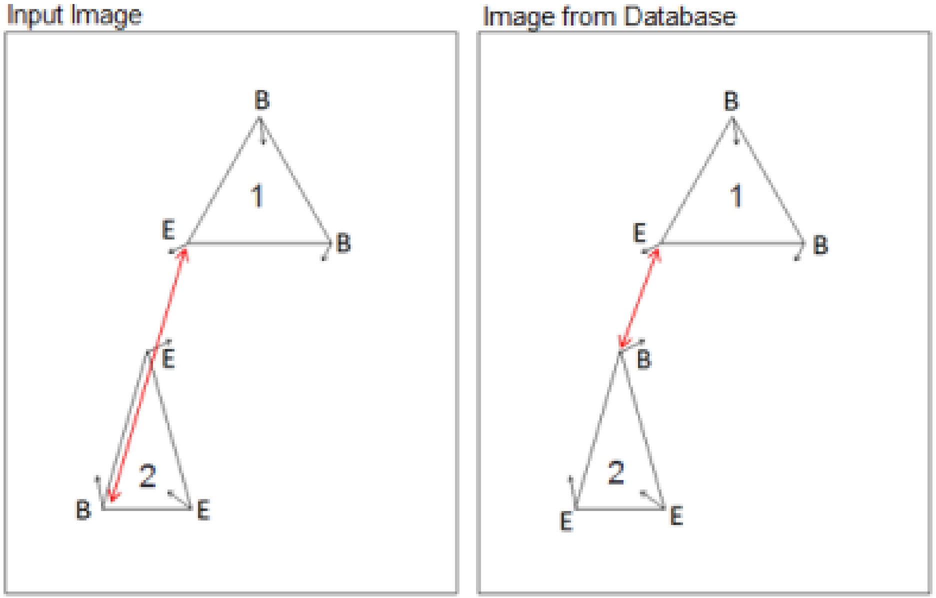


Figure 3.10. Illustrating comparison of unique minutia type distances. ‘B’ represents a bifurcation minutia while ‘E’ represents a ridge ending minutia.

Figure 3.10 illustrates how minutia type can be used indirectly as a distinguishing attribute. Minutia type is used in two different ways:

1) In the initial triangle comparison phase, the triangles labeled as ‘1’ were designated as matching pairs and similarly the triangles labeled as ‘2’ were designated as matching pairs. This implies that the triangles passed the “minutia type count” test, which means that the same number of bifurcation and ridge ending minutia were present in each image. That is, since the ‘1’ triangles both had two bifurcations and one ridge ending, and the ‘2’ triangles both had two ridge endings and one bifurcation, they fulfilled that matching criterion.

2) Then, during the multi-triangle feature comparison phase, the unique minutia type distance comparison test is performed. This test applies when both of the triangles in the image (such as ‘1’ and ‘2’ shown above) have a unique minutia type for one of its three

vertices (i.e., 1 bifurcation and 2 endings, or 1 ending and 2 bifurcations). For this test, the distance between the minutia points of unique type is taken, for both the reference image and input image, as illustrated above. These distances are compared for similarity. Note that this test is not possible when either of the triangles in a pair consist of minutiae that are all of the same type.

Multi-Triangle Test -- Sum and Smallest Inter-Triangle Δ*θ*s for Different Types:

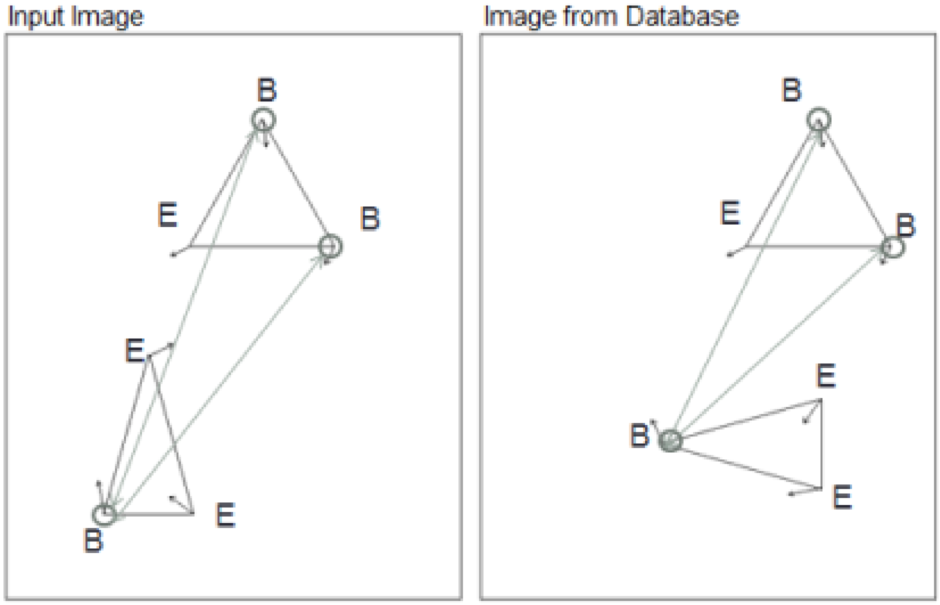


Figure 3.11. Comparing the differences in ridge orientations (Δ*θ*) between minutiae of the same type from different triangles of the feature. ‘B’ represents a bifurcation minutia while ‘E’ represents a ridge ending minutia.

This set of tests helps distinguish cases where one of the triangles in the 2-triangle feature is rotated. That is, at least one of the minutia points might differ and thus its orientation angle will most likely be different. By checking for similarity in differences between orientation angles (Δ*θ*) we are checking for local rotation that may indicate a mismatch of features (see Figure 3.11).

There are three separate checks that must be made: comparisons can be made between bifurcation-bifurcation pairs, ending-ending pairs, or ending-bifurcation pairs, for three total tests in the set. In other words, we separate the bifurcation points from the ending points to devise enhanced matching. Each test checks for similarity between the sum of the $\Delta\theta$ values, and for the smallest $\Delta\theta$ value. If no minutia points of a certain type exist in the triangles being compared, that test obviously will not help distinguish the two sets of triangles. If we ignore minutia type due to its unreliability in being extracted, this test does not apply and the next test to be discussed, “sum and smallest of all inter-triangle $\Delta\theta$s,” will be used instead.

Multi-Triangle Test -- Sum and Smallest of all Inter-Triangle $\Delta\theta$s:

This test is the same as the above except it is used when minutia type is ignored. Instead of having 3 separate $\Delta\theta$ totals (one for each type combination), we have one sum of $\Delta\theta$ total for all inter-triangle minutia pairs. The sum of $\Delta\theta$ total is checked for similarity and the smallest $\Delta\theta$ is checked for similarity.

Single Triangle Test -- Both Minutia Types Required Restriction:

We also can add the restriction to our descriptor that each triangle must have a minutia of both types: ending and bifurcation. This gives our algorithm full discerning power, as when a triangle is composed of all of one type of minutia, the **Inter-Triangle Distances Between Unique Type Minutiae** test cannot be used. Also, when all triangles in the multi-triangle feature are composed of a single type, the **Inter-Triangle $\Delta\theta$s for Different Types** test devolves into the normal inter-triangle $\Delta\theta$ comparison test, adding no additional discerning power.

Multi-Triangle Test – Ridge Count:

We look at the ridge count of all inter-triangle pairwise minutiae and check that the counts match.

3.6 Matching Results

Next we show the results from our matching algorithm. Results were taken using a database of 93 fingerprints from a selection of the FVC2000 DB1 database. Each fingerprint was in turn used as the reference image to be compared against the rest of the database. The reference image is never compared against itself. Thus, all matches are false matches as no true matches exist in the comparisons we make. So the algorithm was developed with the goal of trying to build a highly discriminatory representation of a fingerprint feature matching descriptor with a very low false match rate. The number of false matches could be reduced to zero in many cases if we were looking for exact matches. Instead, we must identify matches that are similar, or matching within certain thresholds, to model the fact that even impressions of the same finger differ by a certain degree. Also, the non-linear distortion present in many latent prints must be accounted for. We give results at given distortion tolerance parameters and also vary those parameters to show the effect those parameters have on the number of matches recorded. We also give results that show the usefulness of each of the different matching tests. Table 3.2 shows the results.

Matching Ratio Calculation Details

In Table 3.2 a column for “Matching Ratio” is provided. This section explains what “Matching Ratio” is and how it is calculated. “Matching Ratio” is the number of total multi-triangle feature matches (from the previous column) divided by an estimate of the total number of possible match comparisons that could have been made (match candidates). It is used to obtain a fairer comparison between 2-triangle feature matches and 3-triangle feature matches (since many more matching possibilities exist when forming 3-triangle sets rather than 2-triangle pairs), and between cases where all triangles are considered and only triangles containing both minutia types are considered.

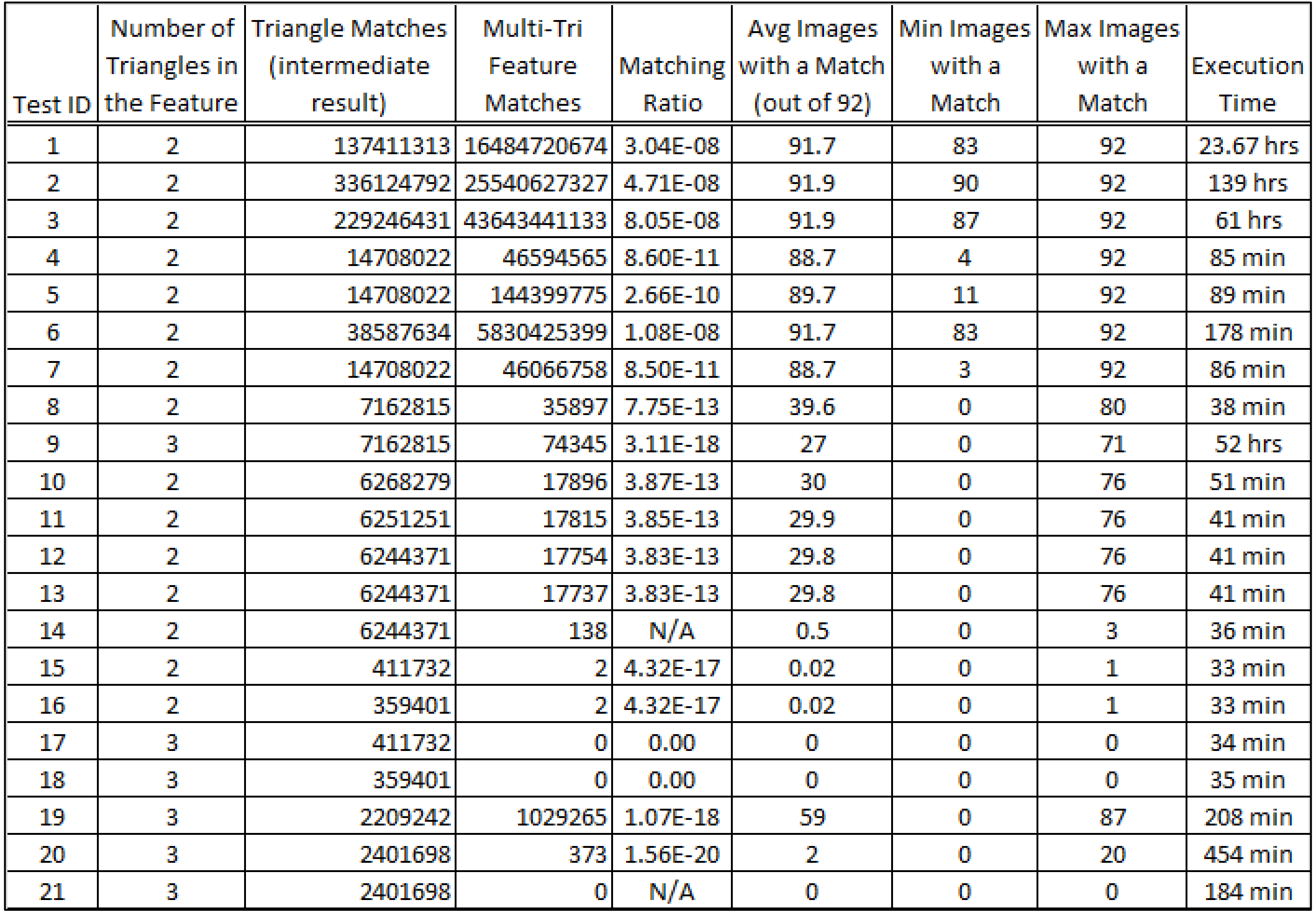
The estimate for total possible match comparisons was made as follows:

It was observed that 371,190 triangles existed in the database (all 93 fingerprints combined) that fell within the maximum side length and minimum side length restrictions. This results in an average of $317{,}190/93 \cong 3991$ triangles per fingerprint. When forming 2-triangle features, on the average, the 3991 triangles result in $C_2^{3991} = 7{,}962{,}045$ possible 2-triangle combinations. Each of the 7,962,045 2-triangle features of the reference fingerprint would be compared to the 7,962,045 2-triangle features of a fingerprint from the database for a one-to-one comparison, for a total of $7{,}962{,}045 \times 7{,}962{,}045 \cong 6.34 \times 10^{13}$ 2-triangle feature comparisons between the two images. This is multiplied by 92 for the 92 one-to-one comparisons that occur for the whole database for one reference image. That result is multiplied by 93 since each of the 93 images takes a turn being the reference image being compared to the 92 other images. We obtain an estimated total number of possible match comparisons of $6.34 \times 10^{13} \times 92 \times 93 \cong 5.42 \times 10^{17}$. Similarly, the total for 3-triangle features could be calculated using $C_3^{3991} = 10{,}586{,}865{,}835$ possible 3-triangle combinations for each image. Also, we can calculate totals in the same way for the case when we restrict that all triangles must contain both minutia types, based on the observation that 200,596 triangles exist in the database containing both types of triangles (2 bifurcations and 1 ending or 1 bifurcation and 2 endings).

In the case where the feature has the restriction that a shared-ridge segment is required, it is hard to make an accurate estimate for the total number of matching comparisons possible, so we do not attempt to determine a “Matching Ratio” in this case.

For reference, we note that a theoretical database of 93 images that were all identical for a comparison of 2-triangle features using these calculations would have a “Matching Ratio” of 1.26×10^{-7}.

Table 3.2. Matching Results



Test ID	Number of Triangles in the Feature	Triangle Matches (intermediate result)	Multi-Tri Feature Matches	Matching Ratio	Avg Images with a Match (out of 92)	Min Images with a Match	Max Images with a Match	Execution Time
1	2	137411313	16484720674	3.04E-08	91.7	83	92	23.67 hrs
2	2	336124792	25540627327	4.71E-08	91.9	90	92	139 hrs
3	2	229246431	43643441133	8.05E-08	91.9	87	92	61 hrs
4	2	14708022	46594565	8.60E-11	88.7	4	92	85 min
5	2	14708022	144399775	2.66E-10	89.7	11	92	89 min
6	2	38587634	5830425399	1.08E-08	91.7	83	92	178 min
7	2	14708022	46066758	8.50E-11	88.7	3	92	86 min
8	2	7162815	35897	7.75E-13	39.6	0	80	38 min
9	3	7162815	74345	3.11E-18	27	0	71	52 hrs
10	2	6268279	17896	3.87E-13	30	0	76	51 min
11	2	6251251	17815	3.85E-13	29.9	0	76	41 min
12	2	6244371	17754	3.83E-13	29.8	0	76	41 min
13	2	6244371	17737	3.83E-13	29.8	0	76	41 min
14	2	6244371	138	N/A	0.5	0	3	36 min
15	2	411732	2	4.32E-17	0.02	0	1	33 min
16	2	359401	2	4.32E-17	0.02	0	1	33 min
17	3	411732	0	0.00	0	0	0	34 min
18	3	359401	0	0.00	0	0	0	35 min
19	3	2209242	1029265	1.07E-18	59	0	87	208 min
20	3	2401698	373	1.56E-20	2	0	20	454 min
21	3	2401698	0	N/A	0	0	0	184 min

Test ID List

The following list identifies the parameters and tests used for each Test ID from Table 3.2. The maximum side length for a triangle was always 150 pixels and the minimum side length for a triangle was always 11. No overlapping minutiae were allowed between triangles, so 2-triangle features are strictly 6 minutiae and 3-triangle features are strictly 9 minutiae. In the following list, “using type” or “not using type” is a parameter which refers to whether the minutia type (differentiating between bifurcations and ridge endings) was considered or ignored. “Centroid distances” is a parameter that refers to using the distance between triangle centroids to determine spatial differences between triangles. “Pairwise distances” is the alternative parameter to centroid distances to determine spatial differences between triangles, which involves the computation of all pairwise distances of minutiae between triangles.

1. 2-triangle feature, distance tolerance 10%, angle tolerance 10°, using type, centroid distances
2. 2-triangle feature, distance tolerance 15%, angle tolerance 10°, using type, centroid distances
3. 2-triangle feature, distance tolerance 10%, angle tolerance 14°, using type, centroid distances
4. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances
5. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, pairwise distances
6. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, not using type, centroid distances

7. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae test”
8. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction
9. 3-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction
10. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests
11. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”
12. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”
13. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using

both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”, using “inter-triangle shared ridge minutiae Δθ test”

14. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”, using “inter-triangle shared ridge minutiae Δθ test”, shared-ridge segment required
15. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using Ridge Count
16. 2-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”, using “inter-triangle shared ridge minutiae Δθ test”, using Ridge Count
17. 3-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using Ridge Count

18. 3-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”, using “inter-triangle shared ridge minutiae Δθ test”, using Ridge Count
19. 3-triangle feature, distance tolerance 5%, angle tolerance 6°, not using type, centroid distances, using Ridge Count
20. 3-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”, using “inter-triangle shared ridge minutiae Δθ test”, using Ridge Count with uncertainty of ±1 (see explanation in Analysis of Results below)
21. 3-triangle feature, distance tolerance 5%, angle tolerance 6°, using type, centroid distances, using “Inter-Triangle Distances Between Unique Type Minutiae” test, using both types required restriction, using shared-ridge count and distances between inter-triangle shared-ridge minutiae tests, using “single-tri shared ridge minutiae Δθ test”, using “distance between intra-tri shared-ridge minutiae test”, using “inter-triangle shared ridge minutiae Δθ test”, shared-ridge segment required, using Ridge Count with uncertainty of ±1 (see explanation in Analysis of Results below)

Analysis of Matching Results (refer to Test ID list)

Test 1 was our initial base test result with reasonable tolerances. Tests 2 and 3 show the massive effects of increased distortion tolerances when the distance and angle tolerances were increased to 15% and 7°, respectively. Note the huge increase in execution time. In order to obtain results in a reasonable amount of time, the tolerances were decreased to 5% distance tolerance and 6° angle tolerance for the remainder of tests, starting with Test 4, which became our new base/reference case.

Test 5 substitutes the centroid distance comparison with the pairwise distance comparison. There is a large increase in the number of (false) matches using the pairwise distance comparison, proving the centroid distance comparison to be a much better distinguisher of triangle spatial differences. The centroid distance comparison also resulted in a faster execution time than using pairwise distance comparison. Thus we use the centroid distance comparison in the remainder of the tests.

In Test 6 we ignore minutia type, meaning we don't differentiate between bifurcations and ridge endings (realistically, extraction tools have a difficult time doing this anyway), so we can't match triangles based on type counts. Also, instead of comparing "Sum and Smallest Inter-Triangle $\Delta\theta$s for Different Types" we can only compare "Sum and Smallest of all Inter-Triangle $\Delta\theta$s". This results in a large increase in triangle matches, which produces a large increase in features matched, as well as a large negative effect on overall execution time since many more 2-triangle features must be checked.

In Test 7 and subsequent tests we again consider minutia type. Test 7 is equivalent to Test 4 except Test 7 includes the "Inter-Triangle Distances Between Unique Type Minutiae test." Results show some improvement in false matches with a reduction of around 500,000.

Test 8 adds the restriction that all triangles must contain both types of minutiae. This allows the type-based tests “Sum and Smallest Inter-Triangle $\Delta\theta$s for Different Types” and “Inter-Triangle Distances Between Unique Type Minutiae” to have maximum effect. There is a very large decrease in (false) matches due to both higher discriminatory power and less triangles being considered. The decrease in the “Matching Ratio” shows the true value of this test.

Test 9 is the first test thus far to use the 3-triangle feature. Results show a large decrease in the “Matching Ratio” as compared to the 2-triangle equivalent (Test 8), giving us **increased confidence in our matches as the number of points in the feature is increased**. Also, the average number of images with a match drops from 39.6 to 27. However, too many false matches exist to make this matching descriptor very useful. Thus, we **need to incorporate extended ridge information.**

Test 10 is the first test to incorporate some extended ridge information, using the idea of shared ridge segments. It reduces the number of (false) matches by about 50%. A small performance penalty is paid due to extra distance calculations (distance of inter-triangle shared ridge segments).

Test 11 (using intra-triangle shared ridge minutiae $\Delta\theta$ comparison) offers only a small reduction in matches reduced but also offers an execution time reduction of almost 20% due to the reduction in triangle matches and subsequent reduction in 2-triangle feature checks that must be performed.

Test 12 shows the benefit (or lack thereof) of comparing the distances of triangle side lengths that are shared-ridge segments. It offers a small benefit (compare to Test 11) with virtually no performance penalty since no additional distance calculations are needed. If the shared-ridge segment is also the smallest side of the triangle, this test is redundant since the smallest sides are already directly compared.

Test 13 shows the added benefit (or lack thereof) of comparing the $\Delta\theta$ values between inter-triangle minutiae that shared a ridge segment. Since the sum of $\Delta\theta$ values are already being compared and this is a subset of that, only a 0.1% false match reduction is achieved.

Test 14 places the restriction that there must be a shared-ridge segment between at least a pair of the 6 minutiae of the feature. We see that this restriction narrows the search space considerably. An average of only 0.5 images out of 92 contained a match over the 93 reference files, while the maximum number of images matched from any one reference file was 3.

Test 15 introduces the Ridge Count matching attribute and shows its great discriminatory power. Without using any shared-ridge information, only 2 matches were found throughout the whole database!

In Test 16 shared-ridge information was considered but it was not able to eliminate the last two false matches.

In Test 17 we use a 3-triangle feature, using Ridge Count and no shared-ridge information, and no false matches were found!

Test 18 can be compared to Test 16, as it is the same except using a 3-triangle feature instead of a 2-triangle feature. We see that with the 3-triangle feature the 2 false matches occurring on Test 16 no longer occur.

Tests 19, 20, and 21 address extraction constraints that may have to be dealt with in real-world applications. As stated previously, minutia type is often difficult to determine, so in Test 19 we attempt to match using a 3-triangle feature using Ridge Count but ignoring type (also shared-ridge information was not used to show the full effect of ignoring minutia type). Over a million false matches were found and the Matching Ratio suffered. Execution time also increased drastically.

Tests 20 and 21 consider that the ridge count extraction may not be completely accurate in any given extraction tool and so for every ridge count extracted it has an uncertainty of ±1 ridge. Using all type comparisons and shared-ridge information along with ridge count with uncertainty ±1, an average of 2 images from the database matched every reference file. However, when we require that each 3-triangle feature must have a shared-ridge segment, zero false matches are found!

By comparing all the results in Table 3.2, we get an idea of the effects that the different parameters have on matching results. We learned that some of the matching attributes we came up with are not very useful additions to the matching algorithm. We see that spatial and angular distortion tolerance parameters greatly affect the ability of the matching algorithm. We see that the minutia information of $[x,y,\theta]$ is not sufficient and that our extended ridge information is useful for matching and can be used with a number of minutiae to provide a certain level of confidence for the quality of the image. Although a feature with 9 minutiae is generally stronger (higher discriminatory power) than a feature with 6 minutiae, it can be observed that there are cases where a 6-minutiae feature can have higher discriminatory power due to other attributes being used. For example, compare Tests 15 and 16 to Test 19. Tests 15 and 16 result in an average of 0.2 images with a match[3], whereas Test 19 results in an average of 59 images with a match. This is because of the fact that minutia type was ignored in Test 19. This speaks to the power of minutia type as a discriminatory attribute, if type extraction can be trusted (which we assume to be the case for the purpose of this work).

[3] Remember, avg. number of images with a match is the avg. over 93 reference cases, where each reference case involves a comparison to 92 images in the database.

Chapter 4: **Mining for Distinctive Features**

4.1 Overview

Using a second method to accomplish our objectives, we mine for “rare” features. By “rare” we mean those features that are distinctive, or statistically occur lower than some threshold. We believe that there exist many such distinctive features that relate minutiae and friction ridges. We build a set of features that are observed to occur infrequently (only one occurrence within our database). The usefulness of such features then can be tested. First, using only the minutiae involved in the feature, we perform a standard minutiae matching algorithm on a small database of fingerprints. Either a set of matches is returned or a ranking score for the most similar fingerprints is returned. Second, we match with knowledge of the distinctiveness of the feature, i.e., we check if that feature exists in the database. Ideally that feature will only be found on a fingerprint that is a true match.

4.2 Database Profiling

In order to choose the correct parameters to narrow the search space to find distinctive features, first we profile the data in our fingerprint database to get an idea of what will constitute “rare”. We create histograms to profile the data. The data we profile for triangles: sum of triangle side lengths, longest side length, shortest side length, sum of ridge counts, number of shared-ridge segments within the triangle (0-3 possible), length of those shared-ridge segments, etc. We can then see that, for example (with max side length restricted to 150 pixels), only 0.065% of triangles have a ridge count > 45 (sample size: 44 fingerprints) (see Figure 4.1). This type of information will be useful later to combine different parameters to find distinctive spatial distributions of minutiae.

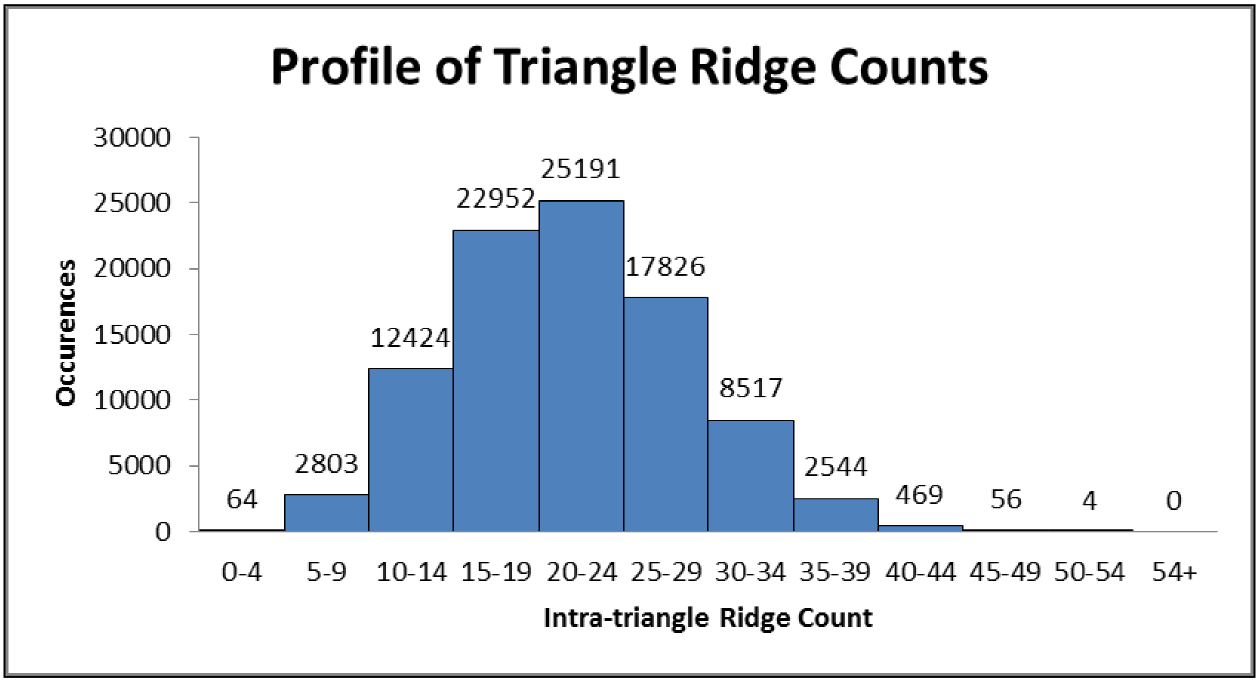


Figure 4.1. Profile of triangle ridge counts

4.3 Feature Mining Algorithm Details

To discover features that are distinctive spatial distributions of minutiae, the algorithm can be divided into a two-part process. In part 1, all triangles (restricted to some max side length *x*) in the fingerprint are processed. Only triangles that are interesting in some way are kept. By interesting, we mean those that have potential to contribute to a distinctive feature. For example, triangles with a high ridge count to side lengths ratio or triangles with one or more shared-ridge segments are kept.

In part 2, all of the interesting triangles that were kept from part 1 are processed and formed into 2-triangle or 3-triangle features. Again, we have a list of parameters that we can adjust in order to find distinctive features. For example, we might look for all 3-triangle features where each of the triangles has at least one shared ridge segment, there is also a shared-ridge segment between each of the triangles, and on each of those inter-triangle shared-ridge segments the ridge count is at least *w* (see Figure 4.2).

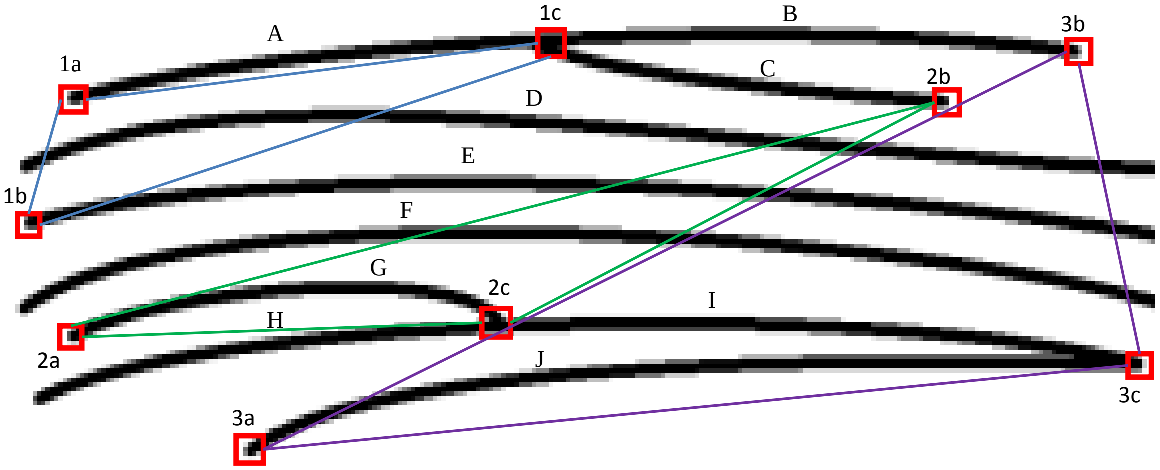


Figure 4.2. Example 9-point feature with triangles 1, 2, and 3 shown on a theoretical partial ridge skeleton. Minutiae are labeled with the number of the triangle they belong to and a lowercase letter identifier. Ridge segments are labeled A-J. Each triangle contains an *intra*-shared-ridge segment (e.g., minutiae 1a and 1c both share ridge segment A), and each triangle is connected to each other via an *inter*-shared-ridge segment (e.g., 1c and 2b share ridge segment C). The rarity of such a feature is even greater when considering the ridge counts between pairwise minutiae, especially if the ridge counts are unusually high or low.

The method for finding such distinctive features involves tuning the parameters such that the feature is unique within our database. Parameters can be set to be more restrictive (less features are returned by the search) or less restrictive (more features are returned by the search). We want to tune the parameters such that a single unique feature is returned by the search. Consider the following example: parameters are selected which yield 5 occurrences of a given feature. Then, we further restrict the parameters, and observe that 0 occurrences are found. So we slightly relax a parameter and 1 occurrence of the feature is found. Since we have found a unique feature within the database, we say such a feature is sufficiently distinctive, rare, or has sufficient discriminatory power. **The importance of this method is not to find definitively rare features or to declare any feature unique, because a slightly different database or different**

extraction tools would yield different results. Our claim is that our methods can be used with any arbitrary extraction tools and any database to find distinctive features to aid in our stated objectives.

4.4 Results: Mined Distinctive Features

The database used was a selection of 93 fingerprints from the FVC2000 DB1 database. Each of the 93 images are from different fingers. The resolution is 500 dpi. We note that our method relies on the fact that the underlying feature extraction is accurate, but is not restricted to any particular extraction technique.

We mined for and discovered a set of 10 distinctive features. (Notes on notation: RC is the notation used for Ridge Count and SR is the notation used for Shared Ridge segment. "sSR" refers to a *short* SR segment, with the requirement that the segment must be shorter than some distance *s*. "tRC" refers to *total* RC, or the sum of intra-RC + inter-RC for the feature. "SL" means *side length* and "SSL" means *sum-of-side-lengths*. All distances are expressed in terms of pixels.)

The distinctive features we found are the following:

1) 2 triangles (6 minutiae), both triangles containing 2 sSR each ($s \leq 31$)
2) 2 triangles (6 minutiae), both triangles with intra-RC ≥ 35, and tRC ≥ 298
3) 2 triangles (6 minutiae), both triangles large in size (SL ≥ 140 for each side), roughly equilateral (since the restriction exists that all side lengths are less than 150; thus the longest side length and the shortest side length could have a maximum difference of 10), and approximately equal RC on each side of the triangle (RC similarity parameter was set so that the RC had to be within 2 on all sides)
4) 3 triangles (9 minutiae), all triangles with intra-RC ≥ 35, tRC ≥ 630

5) 3 triangles (9 minutiae), all triangles large in size (SL ≥ 120 for each side), roughly equilateral (longest side length could only be 8 longer than the shortest), and approximately equal RC on each side of the triangle (RC similarity parameter was set so that the RC had to be within 2 on all sides)
6) 3 triangles (9 minutiae), all triangles have a small SSL to RC ratio (SSL/RC ≤ 17.37) with intra-RC ≥ 19.
7) 3 triangles (9 minutiae), all triangles have a small RC to SSL ratio (SSL/RC ≥ 65.04) with SSL ≥ 100.
8) 3 triangles (9 minutiae), each triangle contains a sSR (s ≤ 60), and each triangle with intra-RC ≥ 39.
9) 3 triangles (9 minutiae), each triangle contains a SR, there is a SR link between each triangle, and tRC ≥ 470.
10) 3 triangles (9 minutiae), each triangle contains a sSR (s ≤ 50), there is a SR link between each triangle, and tRC ≥ 440.

With further exploration of the parameter space, more distinctive features could be found. Figure 4.3 shows an example of a theoretical distinctive feature on an actual fingerprint that is similar to feature #10.

4.5 Verification of Usefulness of Mined Features Via Matching Algorithm

Next, with the mined features, we attempt to show the value of such distinctive features via a feature-based matching algorithm. Considering the minutiae in a latent with only a partial fingerprint image, we perform a triangle-based (rotation- and displacement-invariant) matching against our database. First, we attempt to match based only on the relative distances of the side lengths of the triangles in the feature, and on the differences in ridge orientation between each minutia ($\Delta\theta$ values). For each triangle, the matching parameters are the sum-of-side-lengths, the shortest-side-length, the sum-of-$\Delta\theta$s, and the smallest-$\Delta\theta$. Each triangle in the feature must be matched with a triangle in a potential feature match from the database. Due to non-linear distortion that is typically present in latent fingerprints, the side length and $\Delta\theta$ parameters are considered “matched” when they are sufficiently close in value (e.g., within 10% difference). Then, we consider both the relative distances between the centroids of the triangles and the pairwise inter-triangle $\Delta\theta$ values in the multi-triangle feature. If these centroid distances and $\Delta\theta$ values also match the candidate from the database, we consider this a multi-triangle match. Results show many false matches exist (since the exact minutiae involved were taken from one particular instance), even though the latent contains a distinctive feature.

Next, when incorporating our multi-triangle features, we show that when also including the ridge information (ridge count and shared ridges) as matching requirements the given feature only occurs once, as we have already discovered during the mining process. Thus we verify that the ridge information is important discriminatory information that helps achieve our stated objectives.

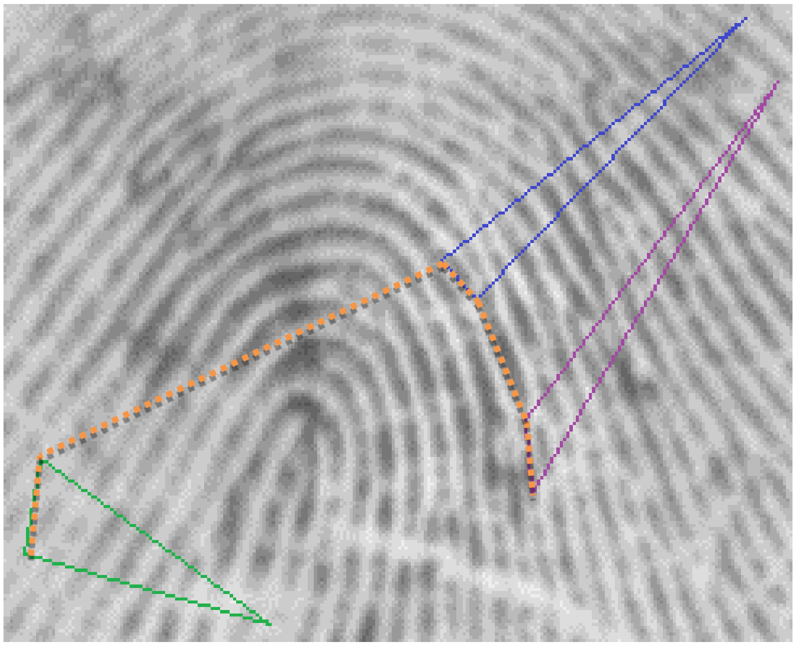
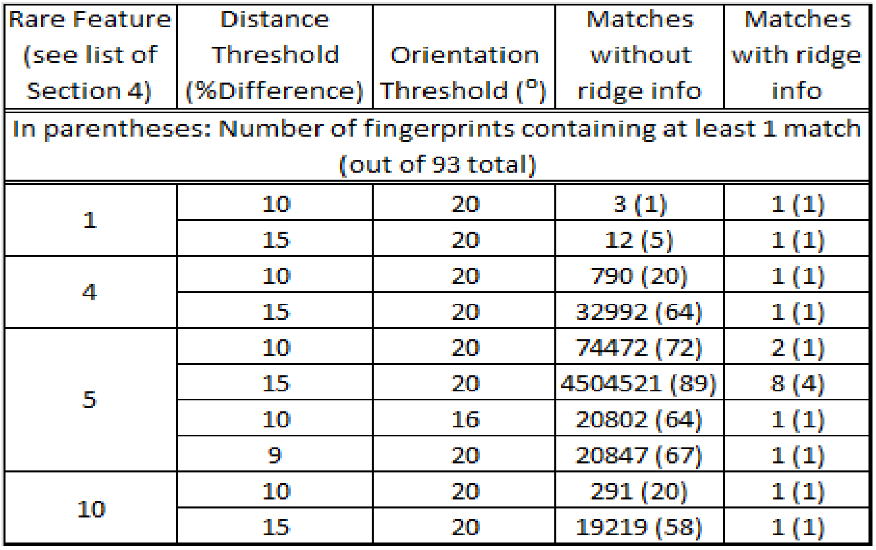


Figure 4.3. Theoretical distinctive feature. Short shared-ridge segments exist on each of the triangles and the total RC is some large value. The shared ridge segments are between the minutiae connected by the dashed lines.

4.6 Verification Results

Table 4.1 shows the results of the matching algorithm when matching the mined features to the database. Results were reported with and without consideration of ridge information. The distance distortion tolerance parameter is set at a given percent difference, meaning any triangle side length or centroid-to-centroid distance comparison is considered matching if within that percent difference. The orientation angle tolerance is nominally set at 20 degrees, meaning any $\Delta\theta$ value comparison within 20 degrees is considered matching. The distinctive features chosen to be verified were chosen due to intuition as to why those features would be rare, and the interesting nature of those features.

Table 4.1. Feature matching with and without ridge info



Rare Feature (see list of Section 4)	Distance Threshold (%Difference)	Orientation Threshold (°)	Matches without ridge info	Matches with ridge info
In parentheses: Number of fingerprints containing at least 1 match (out of 93 total)				
1	10	20	3 (1)	1 (1)
	15	20	12 (5)	1 (1)
4	10	20	790 (20)	1 (1)
	15	20	32992 (64)	1 (1)
5	10	20	74472 (72)	2 (1)
	15	20	4504521 (89)	8 (4)
	10	16	20802 (64)	1 (1)
	9	20	20847 (67)	1 (1)
10	10	20	291 (20)	1 (1)
	15	20	19219 (58)	1 (1)

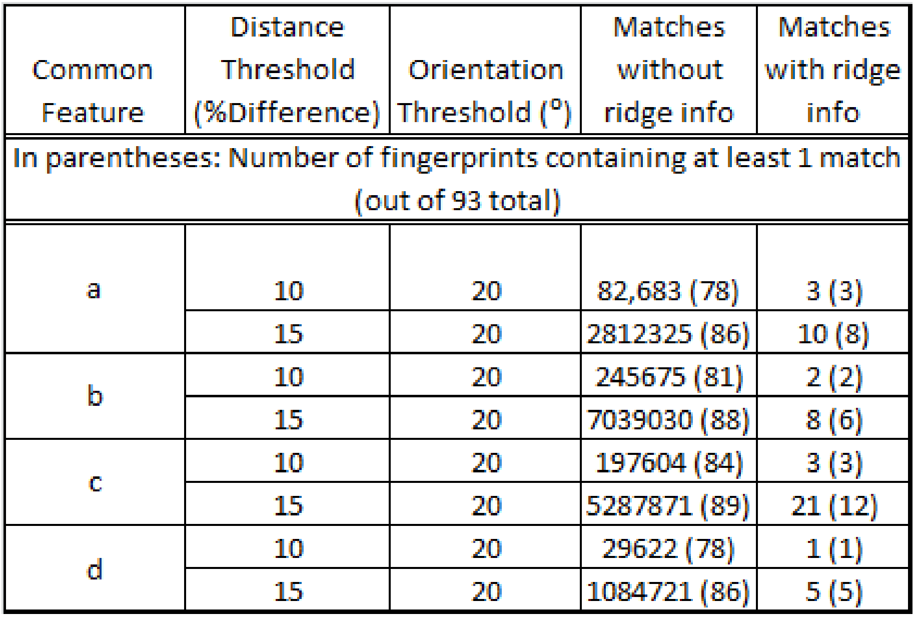
As observed from the table, conventional minutiae-based matching using relative distances and theta values would falsely identify many fingerprints from the database as potential matches. On the other hand, when we incorporate our formulation with ridge information, the number of potential matches is drastically reduced. For example, in feature #4 with distance threshold of 10%, 790 matches were found without ridge information. Some of these matches came from different regions of the same fingerprint image, thus 20 fingerprints were found to have contained a similar layout of the minutia. On the other hand, when ridge information was used, we were able to reduce the number of matches to only 1. In fact, it is the unique match. In only one of the cases shown, feature #5, do false matches exist when matching using ridge information at distance threshold 10% and orientation threshold 20°. This is due to the fact that the formulation of that feature relied on the side lengths of the triangles, which when we match using a distortion tolerance, allows false matches that fall beneath our criteria for rarity. (Also shown in the table for feature #5, if we reduce either the distance or orientation tolerance slightly, the false matches disappear.) Nevertheless, the number of false matches was significantly lower than the case without ridge information.

Table 4.1 also shows that increasing the distortion tolerance greatly increases the number of false matches when not considering the ridge information. For example, in feature #10, the number of potential matches increased from 291 to 19219 when we increase the distance threshold from 10% to 15%! On the other hand, because the ridge information present is so rare, when we include it as a matching parameter the number of matches stays at 1 even when the distance threshold was increased, which is the one true match in the database from which the distinctive feature was originally discovered. This demonstrates the discriminatory power that ridge information adds when performing a match.

Finally, Table 4.2 shows the matching results for several common (non-distinctive) triangle-based features. We use knowledge gleaned from the database profiling (Section 4.2) to know what attributes are common. For example, a typical triangle has a ridge count between 20 and 24, sum-of-side-lengths between 150 and 180 pixels, and no shared-ridge segments. Specific sets of points that exist in the database were chosen, just as the distinctive features were chosen. The difference, in this case, is that even when including the ridge information several false matches appear, unlike in the case of the distinctive features, when usually only the true match exists. For example, in common feature ‘b’ with a distance tolerance of 10%, nearly a quarter of a million matches were found. On the other hand, when ridge information was added, only two matches were returned. This is a tremendous reduction in the number of false matches. Hence, we see that ridge information provides much discriminatory power, and we can significantly reduce the number of false matches. Moreover, if the feature can be categorized as statistically rare, the confidence we have in the match can be increased further.

These methods can be applied to a database of any size to find distinctive features which can help assess the quality of images within that database.

Table 4.2. Matching common features with and without ridge info



Common Feature	Distance Threshold (%Difference)	Orientation Threshold (°)	Matches without ridge info	Matches with ridge info
In parentheses: Number of fingerprints containing at least 1 match (out of 93 total)				
a	10	20	82,683 (78)	3 (3)
	15	20	2812325 (86)	10 (8)
b	10	20	245675 (81)	2 (2)
	15	20	7039030 (88)	8 (6)
c	10	20	197604 (84)	3 (3)
	15	20	5287871 (89)	21 (12)
d	10	20	29622 (78)	1 (1)
	15	20	1084721 (86)	5 (5)

Chapter 5: Conclusions and Future Work

In this thesis, we present two methods for assessing the sufficiency in terms of quality and quantity of information present in a fingerprint image, especially for latent fingerprints. Both methods rely on the same framework: composing triangles from all possible minutiae triplets, and from those triangles hierarchically and modularly considering possible 2- or 3-triangle features. We propose adding extended ridge information to our minutiae triangle features: shared-ridge segments and ridge count (ridges intersected between pairwise minutiae).

In Chapter 3 we apply our minutiae triangle features to a fingerprint matching scheme, with the goal of developing a matching descriptor that has sufficient discriminatory power to eliminate most or all false matches. We allow identification of match candidates to another fingerprint of the same finger within some necessary distortion tolerances. Not only does this scheme have benefits of identifying sufficiency of information, but a scheme such as this could be used in an AFIS application to narrow down the candidate match search space.

In Chapter 4 we propose a method to find distinctive features of a fingerprint that have high discriminatory power within a given database. The features are based on rare spatial distributions of minutiae. We characterize the spatial differences between minutiae and minutiae triangles based on linear distance between points, ridges intersected between points (ridge count), and points connected by shared ridge segments. Human examiners could use such identified rare features for verification or support of their decisions, or to quantify the significance of a match between two fingerprints in a more definitive way.

There is much future work to be done to build off of the work of this thesis.

First of all, the process of finding rare features (Chapter 4) involved a lot of manually tuning parameters. This process could all be automated. If the process were fully automated we could apply the algorithm to any database to find distinctive features without having to first gain insight into the database. This would speed up the whole process.

Second, it would be helpful to analyze certain regions of a fingerprint to gain additional insight into possible distinctive features. For example, we could perform the distinctive feature extraction process on just the core or delta regions of a fingerprint. In current AFIS sometimes the core regions are ignored because they are thought to be less distinctive. We may be able to learn something about these regions so that they may be more useful for matching.

Third, there are several parameter spaces that have not been varied in this work that could be. For example, we could vary the maximum and minimum side length parameters to obtain a more or less localized version of the triangles. In particular, in this thesis the minimum side length parameter was set to ignore triangles with side lengths of 10 pixels or less. This was mostly to remove the effect of false minutia extraction, since minutiae usually aren’t that close together. The NBIS extraction tool considers a minutia to be false if it is within 16 pixels of another minutia and removes the false one. So, another solution would be to use an extraction tool like NBIS that automatically removes such false minutiae.

Along the same lines as varying different parameters, we could use methods to analyze the distortion present in a given latent print to automatically set our distortion tolerance parameters. These types of methods to analyze quality and distortion in order to select the appropriate parameters already exist to some degree [54, 55]. Applying these methods could prove extremely useful.

Another simple task for future work would be to use a larger database. Due to performance and time considerations from both the extraction tools and the algorithms discussed in this thesis, along with limited availability of fingerprint databases, only a small database of 93 images was used. Any results obtained in the future will be stronger with the use of a larger database.

Lastly, a task to undertake for future work would be to integrate the use of GPGPUs to perform the algorithms developed in this thesis to improve execution time. Even with such a small database, with large tolerance parameters it could take days to execute the algorithms of this thesis using certain parameters or lacking discriminatory attributes that give high discriminatory power. However, our algorithms can easily be reconfigured to achieve greater parallelism. Each image in the database could be processed simultaneously with the use of the multiple concurrent threads that are available with the use of GPGPUs. Even different sets of triangles or multi-triangle features within the same image could be processed in parallel through the use of GPGPUs.

References

[1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition (2nd Edition)*, 2 ed.: Springer, 2009.

[2] H. C. Lee and R. E. Gaensslen, *Advances in Fingerprint Technology, 2 edition*. New York: Elsevier, 2001.

[3] A. Newman, "Fingerprinting's Reliability Draws Growing Court Challenges," *N.Y. TIMES*, April 7, 2001.

[4] National Institute of Justice, "Forensic Sciences: Review of Status and Needs," U.S. Department of Justice, Report, 1999.

[5] D. R. Ashbaugh, *Quantitative-qualitative friction ridge analysis: an introduction to basic and advanced ridgeology*. Boca Raton, FL: CRC Press, 1999.

[6] M. R. Hawthorne, *Fingerprints: Analysis and Understanding*: CRC Press, 2009.

[7] National Research Council of the National Academies, "Strengthening Forensic Science in the United States: A Path Forward," The National Academies Press, Washington, DC, 2009.

[8] S. Pankanti, S. Prabhakar, and A. K. Jain, "On the Individuality of Fingerprints," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 24, pp. 1010-1025, 2002.

[9] J. Chen and Y.-S. Moon, "A Minutiae-based Fingerprint Individuality Model," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, 2007, pp. 1-7.

[10] National Institute of Justice, "Forensic Friction Ridge (Fingerprint) Examination Validation Studies," U.S. Department of Justice, Solicitation, 2000.

[11] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Image mosaicing for rolled fingerprint construction," in *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, 1998, pp. 1651-1653 vol.2.

[12] "Electronic Fingerprint Transmission Specification," U.S. Dept. of Justice, Federal Bureau of Investigation, Criminal Justice Information Services Division, [Washington, D.C.] :1999. Available at http://www.doj.state.wi.us/les/CCH/efts_70.pdf.

[13] N. B. Nill, "Test Procedures for Verifying IAFIS Image Quality Requirements for Fingerprint Scanners and Printers," *MITRE Tech. Report: MTR 05B0000016*, April 2005.

[14] A. Ross and A. Jain, "Biometric Sensor Interoperability: A Case Study in Fingerprints," in *Biometric Authentication*. vol. 3087, D. Maltoni and A. Jain, Eds., ed: Springer Berlin / Heidelberg, 2004, pp. 134-145.

[15] R. Arun and N. Rohan, "A calibration model for fingerprint sensor interoperability," in *SPIE Conference on Biometric Technology for Human Identification III*, Orlando, USA, 2006, pp. 62020B-1 - 62020B-12.

[16] A. Ross and R. Nadgir, "A Thin-Plate Spline Calibration Model For Fingerprint Sensor Interoperability," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 20, pp. 1097-1110, 2008.

[17] J. Jang, S. Elliott, and H. Kim, "On Improving Interoperability of Fingerprint Recognition Using Resolution Compensation Based on Sensor Evaluation," in *Advances in Biometrics*. vol. 4642, S.-W. Lee and S. Li, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 455-463.

[18] C. Champod, C. J. Lennard, P. Margot, and M. Stoilovic, "Fingerprint Detection Techniques," in *Fingerprints and Other Ridge Skin Impressions*, ed: CRC Press, 2004.

[19] U. S. Dinish, Z. X. Chao, L. K. Seah, A. Singh, and V. M. Murukeshan, "Formulation and implementation of a phase-resolved fluorescence technique for latent-fingerprint imaging: theoretical and experimental analysis," *Appl. Opt.*, vol. 44, pp. 297-304, 2005.

[20] W.-C. Lin and R. C. Dubes, "A review of ridge counting in dermatoglyphics," *Pattern Recognition*, vol. 16, pp. 1-8, 1983.

[21] X. Jiang and W.-Y. Yau, "Fingerprint minutiae matching based on the local and global structures," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 2000, pp. 1038-1041 vol.2.

[22] L. Sha, F. Zhao, and X. Tang, "Minutiae-based Fingerprint Matching Using Subset Combination," presented at the Proceedings of the 18th International Conference on Pattern Recognition - Volume 04, 2006.

[23] D. A. Stoney and J. I. Thornton, "A Critical Analysis of Quantitative Fingerprint Individuality Models," *Journal of Forensic Sciences*, vol. 31, pp. 1187-1216, 1986.

[24] A. K. Jain, S. Pankanti, S. Prabhakar, L. Hong, and A. Ross, "Biometrics: A Grand Challenge," in *Proc. Int. Conf. on Pattern Recognition (17th)*, 2004, pp. 935-942.

[25] N. Polikarpova, "On the fractal features in fingerprint analysis," in *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, 1996, pp. 591-595 vol.3.

[26] M. Takeda, S. Uchida, K. Hiramatsu, and T. Matsunami, "Finger image identification method for personal verification," in *Pattern Recognition, 1990. Proceedings., 10th International Conference on*, 1990, pp. 761-766 vol.1.

[27] A. Ceguerra and I. Koprinska, "Automatic Fingerprint Verification Using Neural Networks," presented at the Proceedings of the International Conference on Artificial Neural Networks, 2002.

[28] A. V. Ceguerra and I. Koprinska, "Integrating Local and Global Features in Automatic Fingerprint Verification," in *16th International Conference on Pattern Recognition (ICPR'02)*, 2002, pp. 30347-30347.

[29] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *Image Processing, IEEE Transactions on,* vol. 9, pp. 846-859, 2000.

[30] X. Xie, F. Su, and A. Cai, "Ridge-Based Fingerprint Recognition," in *Advances in Biometrics*. vol. 3832, D. Zhang and A. Jain, Eds., ed: Springer Berlin / Heidelberg, 2005, pp. 273-279.

[31] J. Feng, Z. Ouyang, and A. Cai, "Fingerprint matching using ridges," *Pattern Recognition,* vol. 39, pp. 2131-2140, 2006.

[32] J. Feng and A. Cai, "Fingerprint Representation and Matching in Ridge Coordinate System," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, 2006, pp. 485-488.

[33] A. R. Roddy and J. D. Stosz, "Fingerprint features-statistical analysis and system performance estimates," *Proceedings of the IEEE,* vol. 85, pp. 1390-1421, 1997.

[34] B. Bhanu and X. Tan, "A Triplet Based Approach for Indexing of Fingerprint Database for Identification," in *Audio- and Video-Based Biometric Person Authentication*. vol. 2091, J. Bigun and F. Smeraldi, Eds., ed: Springer Berlin / Heidelberg, 2001, pp. 205-210.

[35] B. Bhanu, "Fingerprint Indexing Based on Novel Features of Minutiae Triplets," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, pp. 616-622, 2003.

[36] G. Parziale and A. Niel, "A Fingerprint Matching Using Minutiae Triangulation," in *Biometric Authentication*. vol. 3072, D. Zhang and A. K. Jain, Eds., ed: Springer Berlin / Heidelberg, 2004, pp. 1-50.

[37] X. Chen, J. Tian, X. Yang, and Y. Zhang, "An algorithm for distorted fingerprint matching based on local triangle feature set," *Information Forensics and Security, IEEE Transactions on,* vol. 1, pp. 169-177, 2006.

[38] W. Xu, X. Chen, and J. Feng, "A Robust Fingerprint Matching Approach: Growing and Fusing of Local Structures," in *Advances in Biometrics*. vol. 4642, S.-W. Lee and S. Li, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 134-143.

[39] National Institute of Standards and Technology, "User's Guide to Export Controlled Distribution of NIST Biometric Image Software," ed, 2004.

[40] L. Xiping, T. Jie, and W. Yan, "A minutiae matching algorithm in fingerprint verification," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 2000, pp. 833-836 vol.4.

[41] J. Chen and Y.-S. Moon, "The statistical modelling of fingerprint minutiae distribution with implications for fingerprint individuality studies," in *Computer Vision and Pattern Recognition 2008*, Anchorage, AK, 2008, pp. 1-7.

[42] Y. Zhu, S. C. Dass, and A. K. Jain, "Statistical Models for Assessing the Individuality of Fingerprints," *Information Forensics and Security, IEEE Transactions on*, vol. 2, pp. 391-401, 2007.

[43] C. Neumann, C. Champod, R. Puch-Solis, N. Egli, A. Anthonioz, D. Meuwly, and A. Bromage-Griffiths, "Computation of Likelihood Ratios in Fingerprint Identification for Configurations of Three Minutiæ," *Journal of Forensic Sciences,* vol. 51, pp. 1255-1266, 2006.

[44] M. Tico and P. Kuosmanen, "Fingerprint matching using an orientation-based minutia descriptor," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 25, pp. 1009-1014, 2003.

[45] J. Cheng, J. Tian, and H. Chen, "Fingerprint Minutiae Matching with Orientation and Ridge," in *Biometric Authentication*. vol. 3072, D. Zhang and A. K. Jain, Eds., ed: Springer Berlin / Heidelberg, 2004, pp. 1-2.

[46] C. Hui, Y. Jianping, S. Xin, H. Chunfeng, and L. Yong, "An orientation-based ridge descriptor for fingerprint image matching," in *Computer Design and Applications (ICCDA), 2010 International Conference on*, 2010, pp. V1-288-V1-292.

[47] N. K. Ratha, K. Karu, C. Shaoyun, and A. K. Jain, "A real-time matching system for large fingerprint databases," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 18, pp. 799-813, 1996.

[48] G. Bebis, T. Deaconu, and M. Georgiopoulos, "Fingerprint identification using Delaunay triangulation," in *Information Intelligence and Systems, 1999. Proceedings. 1999 International Conference on*, 1999, pp. 452-459.

[49] H. Deng and Q. Huo, "Minutiae Matching Based Fingerprint Verification Using Delaunay Triangulation and Aligned-Edge-Guided Triangle Matching," in *Audio- and Video-Based Biometric Person Authentication*. vol. 3546, T. Kanade, A. Jain, and N. Ratha, Eds., ed: Springer Berlin / Heidelberg, 2005, pp. 357-372.

[50] C. Wang, "Delaunay Triangulation Algorithm for Fingerprint Matching," in *3rd International Symposium on Voronoi Diagrams in Science and Engineering (ISVD'06),* 2006, pp. 208-216.

[51] Y. Yin, H. Zhang, and X. Yang, "A method based on Delaunay triangulation for fingerprint matching," in *SPIE Conference on Biometric Technology for Human Identification II*, Orlando, FL, USA, 2005, pp. 274-281.

[52] L. O’Gorman, "Fingerprint Verification," in *Biometrics*, A. K. Jain, R. Bolle, and S. Pankanti, Eds., ed: Springer US, 2002, pp. 43-64.

[53] S. Prabhakar, A. K. Jain, and S. Pankanti, "Learning fingerprint minutiae location and type," *Pattern Recognition,* vol. 36, pp. 1847-1857, 2003.

[54] L. Shen, A. Kot, and W. Koo, "Quality Measures of Fingerprint Images," in *Audio- and Video-Based Biometric Person Authentication*. vol. 2091, J. Bigun and F. Smeraldi, Eds., ed: Springer Berlin / Heidelberg, 2001, pp. 266-271.

[55] Y. He, J. Tian, Q. Ren, and X. Yang, "Maximum-Likelihood Deformation Analysis of Different-Sized Fingerprints," in *Audio- and Video-Based Biometric Person Authentication*. vol. 2688, J. Kittler and M. Nixon, Eds., ed: Springer Berlin / Heidelberg, 2003, pp. 1062-1062.