CHAPTER 3

SEGMENTATION BASED ENHANCED FINGERPRINT RECOGNITION

3.1 INTRODUCTION

Fingerprints are widely used in biometric science for the validation and verifications of entry into specific task that needs to be more reliable, efficient and accurate. Fingerprint recognition or fingerprint authentication refers to the automated method of verifying an individual identity for authentication purpose. Fingerprints are one of many forms of biometrics used to identify individuals and verify their identity. The fingerprint area contains the information that may be useful in many fingerprint processing stages, while the rest of the image might encompass relatively useless information and unwanted signals, such as noise. The unwanted noises are removed in pre-processing Antonio et al. (2011).

Fingerprint based personal identification has been routinely used in forensic laboratories and identification units around the world and it has been accepted by the courts of law which use it as a method of evidence for nearly a century. The fundamental premises on which fingerprint identification is based are permanent and fingerprints of an individual are unique Mohammed Saigaa et al. (2014), Maddala et al. (2011). The validity of the first premise has been established based on the anatomy and morphogenesis of
friction ridge skin. The second premise is debatable. The notion of fingerprint uniqueness has been widely accepted, based on manual inspection (by dactyloscopic experts) of millions of fingerprints. The uniqueness problem may be considered as determining the probability that any two individuals may have sufficiently similar fingerprints in a given target population. Gonzaga et al. (2010).

Human experts and Automatic Fingerprint Identification Systems (AFIS) declare that the fingerprints originate from the same source if they are “sufficiently” similar. The degree of similarity depends on typical (intra class) variations observed in multiple impressions of a finger. In order to solve the problem of uniqueness, it is necessary to define the representation of a fingerprint (pattern) and the similarity metric. Fingerprints can be represented by a large number of features including the overall ridge flow pattern, ridge frequency, location and position of singular points (core(s) and delta(s)), type, direction, and location of minutiae points and ridge counts between the pairs of minutiae is shown in the Figure 3.1.

Figure 3.1 Typical features of Fingerprint image
There are two approaches for determining the uniqueness of the fingerprints. In the empirical approach, representative samples of fingerprints are collected and using a typical fingerprint matcher, the accuracy of the matcher on the samples provides an indication of the uniqueness of the fingerprint with respect to the matcher. In the theoretical approach to the estimation of uniqueness, all the realistic phenomena affecting inter class and intra class pattern variations are modelled. The similarity metric is possible to estimate the probability of a false association. The representation of fingerprints minutiae, which is exploited by forensic experts, has been demonstrated to be relatively stable and has been adopted by the majority of automatic fingerprint matching systems. The similarity metric is the number of corresponding minutiae between two minutiae sets as shown in Figure 3.2.

Figure 3.2  Minutiae in Fingerprint

This chapter presents details of fingerprint segmentation and recognition system which consists of different stages, that includes pre-processing, extraction and matching of images from input images using Three Part Decomposition with Position Based (G3PD) method for fingerprint segmentation. The G3PD method decomposes the image into corresponding three parts as Texture, Cartoon and Noise images. The proposed system was
trained and tested using MSU_Veridicom database. The details of fingerprint segmentation using global variation method for segmentation in fingerprints is presented in the following sections.

3.2 LITERATURE SURVEY

Syamala et al. (2013) proposed a faster and an efficient way to remove salt-and-pepper impulse noise and also the edge-preserving regularization of the hence forth obtained fingerprint noise free image using B-Splines. The performance proved better compared with previously proposed nonlinear filters or regularization methods both in terms of noise removal as well as edge regularization for image forensics.

Nainjot Singh et al. (2015) fingerprints are used as input to image fusion mechanism. Daubechies Wavelet transformation is applied on them. Various fusion rules are applied on wavelet coefficients like mean, add, maximum, minimum. Quality of fingerprints are being tested using parameters such as PSNR, Average Difference, Entropy, Chi-Square and proved to be providing the security to the users.

Behna et al. (2014) includes a new method of using directional filter bank for feature extraction, combination of LDA algorithms for dimensionality reduction and Euclidean distance for matching and classification purpose. Also the information fusion is used with combination of fingers to improve the performance of recognition. This work is used at feature level. Poly-U Finger-Knuckle-Print database is used to evaluate the accuracy and found effective.

Mohammed Saigaa et al. (2012), System based on texture of the hand knuckles, namely Finger-Knuckle-Print (FKP), is proposed. To extract
the image local texture information and represent the FKPs features, the 2D Block based Discrete Cosine Transform (2D-BDCT), is employed. Finally, performance of all finger types is determined individually and a min rule fusion is applied to develop a multimodal system. Experimental results show that 2D-DCT-mod2 yields best performance for identifying FKPs and it is able to provide an excellent recognition rate and provide more security.

Usha et al. (2014) proposed a novel Finger Knuckle Print (FKP) recognition technique based on Haar-Wavelet Transform (HWT). Haar-Wavelet Transform is used to transform the original knuckle image into a subset of its feature space known as `Eigen Knuckle. The principle components and local space variations are extracted and represented in the form of Eigen vectors. Matching of a knuckle images for personal identification is done by means of a classifier using association. Matching scores obtained from various finger knuckles of the same person are fused by means of sum-weighting rule of matching score level fusion. From the exhaustive experiments conducted using two publically available database for FKP, viz. Poly U FKP database and IIT FKP database.

Guangwei et al. (2013) User flexibility in positioning fingers also leads to a certain degree of pose variations in the collected query FKP images. The widely used Gabor filtering based competitive coding scheme is sensitive to such variations, resulting in many false rejections. To improve this problem by reconstructing the query sample with a dictionary learned from the template samples in the gallery set. The reconstructed FKP image can reduce much the enlarged matching distance caused by finger pose variations; however, both the intra-class and inter-class distances will be reduced. Score level adaptive binary fusion rule to adaptively fuse the matching distances before and after reconstruction, aiming to reduce the false rejections without increasing the false acceptances.
Segmentation is an important processing step because, if done correctly, it serves to separate information from noise. During the different fingerprint image processing stages, the background noise may be interpreted as useful information. Various authors focused that removing noises in fingerprint by using wavelet transformations, directional filter banks and textures. It shows that using point pattern, image based and graph based techniques shown index value and higher error rates.

There are limitation relating to the identification of the parameters in decomposition and minimization steps in aforementioned approaches with different illuminations, noises and ghost fingerprints. Addressing these issues the proposed segmentation based fingerprint recognition techniques and their procedure with experimental results are presented below.

3.3 ARCHITECTURE OF SEGMENTATION BASED ENHANCED FINGERPRINT RECOGNITION

Fingerprint segmentation and recognition system consist of automated methods for segmentation and recognizing a person based on the ridge characteristics of the fingerprint images. Segmentation and recognition technologies have become the foundation for an extensive array of highly secure identification and personal verification solutions. The proposed Three Part Decomposition with position based (G3PD) architecture shown in Figure 3.3. It provides a comprehensive view of the fingerprint segmentation and recognition system. Fingerprint image segmentation is an important processing step because, if done correctly, it serves to separate information from noise. During the different fingerprint image processing stages, the background noise may be interpreted as useful information.
Initially the fingerprint recognition starts with image acquisition. The proposed G3PD consists of number of steps such as normalizing the fingerprint image and then creating a template of print to be compared to those in the database Behnam et al. (2014). The G3PD methods Arnold et al. (2004) follows the same philosophy of texture image extraction as the fourier based FDB method Drahansky et al. (2004), Proposed variation model for G3PD Decomposition techniques are at the core of variation methods which is represented in Figure 3.4.

The global three part variation method consists of three parts. These three parts are decomposed in to five elements.

1. **Cartoon:** Anisotropic Total Variation (TV) norm can be used to measure the Piecewise constant regions. Drahansky et al. (2003).

2. **Texture (sparsity):** The sparsity of the texture pattern is measured by the first norm which is well known to enhance the sparseness of the solution.
The sparsity of the texture pattern is measured by the first norm which is well known to enhance the sparseness.

3. **Texture (curve let):** The curve let coefficients of the texture image is enforced by the ‘1’ norm.

4. **Noise:** Noise is measured by the supermoms norm of its curve let coefficients.

5. **Reconstruction constraint:** Finally, the constraint \( f = u+v+\varepsilon \) ensures that the sum of the three component images reconstructs the original image \( f \).

![Diagram of Architecture of Fingerprint Recognition System](image)

**Figure 3.4** Architecture of Fingerprint Recognition System

Figure 3.4 depicts how the image acquisition passes through pre-processing feature extraction, matching process and decision making process. The viable outcome is the process of fingerprint templates.
3.3.1 Image Acquisition

This section introduces a prototype automatic identity authentication system that is capable of authenticating the identity of an individual, using fingerprints Menezes et al. (1996). Fingerprint images are acquired from databases. The acquired fingerprint images may contain various noises, thus cause poor matching results. To remove the noise, the use of Directional Median Filter (DMF) was suggested and used. Boneh et al. (2001).

3.4 FINGERPRINT IMAGE PRE-PROCESSING

A new impulse detector, which is based on the differences between the current pixel and its neighbours aligned with four main directions, has been proposed in Maddala et al. (2011). However in (Bonfig et al. 2004) five steps are proposed to get a high quality or enhanced fingerprint image. These steps are represented in the Figure 3.5.

![Figure 3.5 Fingerprint Pre-processing System](image)

3.4.1 Converting the given grey scale image to binary image (Binarization)

In binarization, the grey scale image is converted into binary image. Binary images are easy to process. The basic principle of converting an image into binary is to decide a threshold value, and then the pixels whose value are more than the threshold are converted to white pixels, and the pixels
whose value are below or equal to the threshold value are converted to black pixels. Mohammed Saigaa, et al. (2012)

The threshold value has been decided using Otsu method. Threshold value of a small window (10 * 10) of the image can be calculated for calculating the threshold of the entire image. The value can be converted into binary. Then the window is shifted to the next position and binarization is done. In this way the entire image is converted to binary.

3.4.2 Central line thinning of the image.

In central line thinning, after we get the binary image, the next task is to thin the image. The followed central line thinning algorithm produce the thinned image. Drahansky et al. (2001). Central line thinning algorithm produces better result than other thinning algorithm because the basic structure and alignment of the ridges remain same after thinning as in the original image. There are total 23 templates defined for thinning algorithm. By referring to these templates the algorithm decides which black pixels to be converted to white or which black pixels refined. This results in a thinned image. Kumar et al. (2011).

It is easy to develop algorithm for minutiae detection in thinned image. If the width of ridge is more than one pixel, then it is very hard to develop algorithm for minutiae detection. The thinned image has a single pixel width ridges. Guangwei et al. (2012)
3.4.3 Dilation of the thinned image

Dilation is a process to make the given images smoother. All the holes are filled and the edges are smoothed in dilation. A structuring element has been used for dilation which is represented in the Figure 3.6

![Structuring Element used for dilation](image)

**Figure 3.6 Structuring Element used for dilation**

The dimension of the structuring element is $3 \times 3$. It is moved on the image sequentially and if any one of the branches coincides with black pixel, the central pixel is converted to black. Otherwise central pixel will be white. Usha et al. (2014)

3.4.4 Removing unwanted portions from the image (Refining)

There are many small unwanted portions that are unnecessary for further processing, but these portions if exist may lead to incorrect minutiae detection. These portions usually consist around 20 to 25 pixels. It is a heuristic approach to decide the number of pixels. The idea is to remove all connected components which have equal to or less than 20 pixels. To remove these first find out all the connected components present in the image. Refining algorithm is used (Cao et al. 2010) to find all the connected components and then removed those portions which consists of less than or equal to 20 pixels. This process is known as Refining of the image.
3.5 FINGERPRINT FEATURE EXTRACTION

The first scientific studies on fingerprint classification were made by (Choi et al. 2011) who divided the fingerprints into three major classes. All the classification schemes are currently used by police agencies, so-called Henry’s classification scheme. As mentioned in the previous section, the uniqueness of a fingerprint is exclusively determined by the local ridge characteristics and their relationships. Carrasco et al. (2010). The ridges and valleys in a fingerprint alternate are flowing in a local constant direction. Eighteen different types of fingerprint features have been enumerated by (Federal Bureau of Investigation, 1984). Further, a total of 150 different local ridge characteristics (islands, short ridges, enclosure, etc.) have been identified by (Kawagoe et al., 1984). These local ridge characteristics are not evenly distributed. Most of them depend heavily on the impression conditions and quality of fingerprints and are rarely observed in fingerprints. The two most prominent local ridge characteristics are: 1) ridge ending and, 2) ridge bifurcation

3.5.1 Edge Detection

A ridge ending is defined as the point where a ridge ends abruptly. A ridge bifurcation is defined as the point where a ridge forks or diverges into branch ridges. Collectively, these features are called minutiae. Most of the fingerprint extraction and matching techniques restrict the set of features to two types of minutiae: ridge endings and ridge bifurcations, as shown in Figure. 3.7 A good quality fingerprint typically contains about 40–100 minutiae.
The Canny Edge Detection algorithm is well known as an optimal edge detector in digital images. The implementation of canny algorithm is done in several steps.

The algorithm runs in 5 separate steps:

**Step 1.** Smoothing: Blurring of the image to remove noise.

**Step 2.** Finding gradients: The edges should be marked where the gradients of the image has large magnitudes.

**Step 3.** Non-maximum suppression: Only local maxima should be marked as edges.

**Step 4.** Double thresholding: Potential edges are determined by thresholding.

**Step 5.** Edge tracking by hysteresis: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge.

### 3.5.1.1 Smoothing

It is inevitable that all images taken from a camera will contain some amount of noise. To prevent that noise mistaken for edges, noise must...
be reduced. Therefore the image is first smoothed by applying a Gaussian filter. The kernel of a Gaussian filter with a standard deviation of $\alpha = 1.4$ is shown in Equation (3.1). The effect of smoothing the test image with this filter is shown in Figure 3.8.

![Fingerprint Edge Smoothing](image)

**Figure 3.8** Fingerprint Edge Smoothing

### 3.5.1.2 Finding Gradient

The Canny algorithm basically finds edges where the grayscale intensity of the image changes the most Dornberger et al. (2003). These areas are found by determining gradients of the image. Gradients at each pixel in the smoothed image are determined by applying what is known as the Sobel operator. Initially to approximate the gradient in the x- and y-direction respectively by applying the kernels. The gradient magnitudes can then be determined as an Euclidean distance measure by applying the law of Pythagoras as shown in Equation (3.1). It is sometimes simplified by applying Manhattan distance measure as shown in Equation (3.2) to reduce the computational complexity. The Euclidean distance measure has been applied to the test image. The computed edge strengths are compared to the smoothed image in Figure 3.8.

$$|G| = \sqrt{G_x^2 + G_y^2}$$  

(3.1)
\[ |G| = |G_x| + |G_y| \]  \hspace{1cm} (3.2)

Where \( G_x \) and \( G_y \) are the gradients in the x- and y-directions respectively.

It is obvious from Figure 3.6, that an image of the gradient magnitudes often indicate the edges quite clearly. However, the edges are typically broad and thus do not indicate exactly where the edges are. To make it possible this direction of the edges must be determined and stored as shown in Equation (3.3).

\[ \theta = \arctan \left( \frac{|G_y|}{|G_x|} \right) \]  \hspace{1cm} (3.3)

Where \( \theta \) is the direction of the edges. \( G_x \) and \( G_y \) are the gradients in the x- and y-directions respectively.

### 3.5.1.3 Non-Maximum Suppression

The purpose of this step is to convert the “blurred” edges in the image of the gradient magnitudes to “sharp” edges. Basically this is done by preserving all local maxima in the gradient image and delete everything else.

The algorithm is for each pixel in the gradient image:

1. Round the gradient direction \( \theta \) to nearest 45°, corresponding to the use of an 8-connected neighbourhood.

2. Compare the edge strength of the current pixel with the edge strength of the pixel in the positive and negative gradient direction. I.e. if the gradient direction is north (\( \theta = 90^\circ \)), compare with the pixels to the north and south.
3. If the edge strength of the current pixel is largest; preserve the value of the edge strength. If not, suppress (i.e. remove) the value.

3.5.1.4 Double Thresholding

The edge-pixels remaining after the non-maximum suppression step are (still) marked with their strength pixel-by-pixel. Many of these will probably be true edges in the image, but some maybe caused by noise or color variations for instance due to rough surfaces. The simplest way to discern between these would be to use a threshold, so that only edges stronger that a certain value would be preserved Syamala et al. (2013). The Canny edge detection algorithm uses double thresholding. Edge pixels stronger than the high threshold are marked as strong and edge pixels weaker than the low threshold are suppressed. Edge pixels between the two thresholds are marked as weak. The effect on the test image with thresholds of 20 and 80. These threshold values can be fixed based on the approximate upper and lower values.

3.5.1.5 Edge Tracking by Hysteresis

Strong edges are interpreted as “certain edges”, can immediately included in the final edge image. Weak edges are included if and only if they are connected to strong edges. The logic is of course that noise and other small variations are unlikely to result in a strong edge (with proper adjustment of the threshold levels). Thus strong edges will (almost) only be due to true edges in the original image. The weak edges can either be due to true edges or noise or color variations. The latter type will probably be distributed independently of edges on the entire image, and thus only a small amount will be located adjacent to strong edges. Weak edges due to true edges are more likely to be connected directly to strong edges. Edge tracking can be implemented by
Binary Large Object (BLOB) analysis. The edge pixels are divided into connected BLOB’s using 8-connected neighbourhood. BLOB’s containing at least one strong edge pixel are then preserved, while other BLOB’s are suppressed.

3.5.2 Ridge Detection and Valley Detection

Ridges and valleys are formed with the points where the intensity gray level reaches a local extreme in a given direction illustrated in Figure 3.9. This direction is the normal to the curve traced by the ridge or respectively the valley at this point Sehasnainjot Singh et al. (2015). Crest lines correspond to important features in many images. Ridges and valleys are attached but not limited to roads in aerial images or blood vessels in medical images.

![Ridges and Valleys Detection](image)

**Figure 3.9  Ridges and Valleys Detection**

3.5.2.1 Difference of Rotated Half Smoothing Filters (DRF)

To estimate at each pixel a smoothed second derivative of the image along a curve crossing this pixel. In one dimension, the second derivative of a signal can be estimated to a DoG operator. By applying two
filters with two different $\lambda$ and the same $\mu$ to obtain directional derivatives. Then, to compute the difference of these two filters to obtain the desired smoothed second derivative information in the thin net directions illustrated in Figure 3.10.(b).

Figure 3.10  (a) A DRF (b) DRF in the thin net directions (c) Discretized filter

3.5.2.2 Pixel Classification

Applying by the convolution of the DRF filter to each pixel of an image. By means of the technique called rotated images, as defined above, to obtain for each pixel a signal which corresponds to a $360/\Delta\theta$ scan in all directions. Then to characterize pixels which belong to a crest line (a ridge or a valley), and thus to build detector. Let $D(x, y, \theta)$ be the pixel signal obtained at pixel $P$ located at $(x, y)$. In the Equation (3.4) $D(x, y, \theta)$ is a function of the direction $\theta$ such that:

$$D(x,y,\theta) = G(\mu,\lambda_1)(x,y,\theta) - G(\mu,\lambda_2)(x,y,\theta)$$  \hspace{1cm} (3.4)

Where $x$ and $y$ are pixel coordinates. $\mu$, $\lambda_1$ and $\lambda_2$ correspond to the standard deviations of the Gaussians. Samples are represented in Figure.3.10. Defines a ridge or valley operator $\Sigma(x, y)$ by the Equation (3.5)
\[ \Sigma(x, y) = D(x, y, \theta_{M1}) + D(x, y, \theta_{M2}) + D(x, y, \theta_{m1}) + D(x, y, \theta_{m2}) \] (3.5)

where \( \theta_{M1}, \theta_{M2} \) are the directions of the local maxima of the function \( D \) and \( \theta_{m1}, \theta_{m2} \) the directions of the local minima. Conditions of detection are as follows:

if \( \Sigma(x, y) > \Sigma_{th} \), the pixel P belongs to a ridge line,

if \( \Sigma(x, y) < -\Sigma_{th} \), the pixel P belongs to a valley line, where \( \Sigma_{th} > 0 \).

Figure 3.11  Examples of functions \( D(x, y, \theta) \)
On a typical valley for example point 1 in Figure 3.11, the pixel signal at the minimum of a valley contains at least two negative sharp peaks. For ridges for example point 7 in Figure 3.11, the pixel signal at the maximum of a ridge contains at least two positive peaks. These sharp peaks correspond to the two directions of the curve entering and leaving path. In case of a junction, the number of peaks corresponds to the number of crest lines in the junction for example point 4 in Figure 3.11. To obtain the same information for bended lines illustrated in point 2 on Figure 3.11. However, at the level of an edge, the absolute value of $\Sigma$ is close to 0 because the absolute value of $D$ at $\theta_{M1}$, $\theta_{M2}$, $\theta_{m1}$ and $\theta_{m2}$ are to each other close for example points 6 and 7 on Figure 3.11. Finally, due to the strong smoothing, $D$ is close to 0 in the presence of noise without neither crest line nor edge illustrated in point 10 in Figure 3.11, Note that $\Sigma_{th}$ can be a parameter for the hysteresis threshold.

3.5.2.3 Ridge and Valley Extractions

**Figure 3.12** (a) $\eta$ extraction ($\Sigma(x, y) > \Sigma_{th}$). (a) $\eta$ computation from $\theta_{M1}$ and $\theta_{M2}$ (b) $\eta$ corresponds to the direction perpendicular to the crest line at the level of a pixel P

Once $\Sigma(x, y)$ computed, then simply estimate $\eta(x, y)$ by Equation (3.6) and (3.7) has shown in the Figure 3.12. (a) and (b)) by:
\[ \eta(x,y) = \frac{\theta M_1 + \theta M_2}{2}, \text{when } \Sigma(x,y) > \Sigma t \] (3.6)

\[ \eta(x,y) = (\theta m_1 + \theta m_2)/2, \text{when } \Sigma(x,y) < -\Sigma th \] (3.7)

Thus, from \( \Sigma(x, y) \) and \( \eta(x, y) \) crest lines can easily be extracted computing local maxima of \( \Sigma(x, y) \) in the direction \( \eta(x, y) \) for ridge detection and the minima for valley detection shown in the Figure 3.13 and Figure 3.14.

Figure 3.13  (a).Original image 500×500 (b) Ridge detection

Figure 3.14  (a) Original image 312×312 (b) Valley detection
3.6 FINGERPRINT MATCHING PROCESS USING GLOBAL VARIATIONAL METHOD

A fingerprint image can be considered as a composition of three components: texture, homogeneous parts and small scale objects. Global Variation Positional Method (G3PD) method aims to decompose a fingerprint image into the corresponding three parts:

**Texture image:** By texture refer to the fact that fingerprint images are highly determined by their oriented patterns which have frequencies only in a specific band in the Fourier spectrum.

**Cartoon image:** The homogeneous regions correspond to the lower frequency response.

**Noise image:** Small scale objects staying in the higher frequency band are considered as noise, e.g. black dots with random position and intensity.

![Segmentation by G3PD method](image)

**Figure 3.15 Segmentation by G3PD method**

For the purpose of fingerprint segmentation interested in the texture image as a feature for segmentation. After the decomposition, the
cartoon and noise images are ignored. Therefore, the decomposition can be considered as a feature extraction step which has the goal to estimate the best possible texture image for a given input image. Subsequently, the region of interest (ROI) is obtained by morphological operations on the non-zero coefficients in the extracted texture image, overall process is shown in Figure 3.15.

3.7 FINGERPRINT DECISION MAKING PROCESS

Decomposition techniques are at the core of variation methods. Decomposition is performed by finding the solution of a convex minimization problem. Global three-part decomposition which has five steps:

Step 1. Cartoon: Piecewise constant regions are measured by the anisotropic total variation (TV) norm.

Step 2. Texture: The sparsity of the texture pattern is measured by the $l_1$ norm which is well-known to enhance the sparseness of the solution.

Step 3. Texture: The smoothness of the texture image is enforced by the $l_1$ norm of the curvelet coefficients.

Step 4. Noise: Noise is measured by the supremum norm of its curvelet coefficients.

Step 5. Reconstruction constraint: Finally, the constraint $f = u + v + \varepsilon$ ensures that the sum of the three component images reconstructs the original image $f$. Empirically, we have found that the curvelets capture the geometry of fingerprint patterns better than classical wavelets. Figure 3.16. Illustrate the effects of the smoothness and sparseness of $v$ after different numbers of iterations.
Figure 3.16 Effects of smoothness and sparseness

The effects of smoothness and sparseness in Figure 3.15 is explained below

Image (a): depicts the original image where the yellow line indicates the boundary of the ROI estimated by the G3PD method after 20 iterations.

Images (b-e): Show $v_{\text{smooth}}$ in (20), smoothing term of $v$.

Image (f): The ROI is obtained using morphological operations on the binarized image.

Image (g-j): visualize the corresponding $v_{\text{update}}$ in (20) after k iteration

3.7.1 Segment Matching

For every minutiae in the image, connect it with its nearest minutiae to form a line segment, and record:
1) Length of this line segment,

2) The direction of this line segment,

3) Number of ridge counts between the two points.

Then, map the line segment by using this rule shown in Figure 3.17.

![Figure 3.17 Line Segment map of a Fingerprint.](image)

Here a new parameter: connection number (CN), which denotes the number of the minutiae points connecting to one minutiae. In Figure 3.18. Shows several possible values of the CN of the minutiae point a.

![Figure 3.18 Values of the CN, (a) CN=1; (b) CN=2; (c) CN=3](image)
For calculating the segment values of $P$ and $Q$ as shown in Equation 3.8.

$$P = \{(x^p_1, y^p_1, \theta^p_1), (x^p_2, y^p_2, \theta^p_2), \ldots, (x^p_M, y^p_M, \theta^p_M)\}$$

$$Q = \{(x^q_1, y^q_1, \theta^q_1), (x^q_2, y^q_2, \theta^q_2), \ldots, (x^q_N, y^q_N, \theta^q_N)\}$$  \(3.8\)

$M$ minutiae in the template and the $N$ minutiae in the input image, respectively. Then, let’s begin to create the line segment vector in Equation (3.9).

$$\text{segment}[i] = \left\{ x_i, y_i, x'_i, y'_i, \text{dist}_i, t_i, t'_i, \text{ang}_i, \text{ridgenum}_i, \text{cn}_i \right\}$$  \(3.9\)

where $x_i$ and $y_i$ denote the coordinates of the $i^{th}$ minutiae in the fingerprint image. $x'$ and $y'$ denote the coordinates of the $i^{th}$ minutiae’s nearest minutiae point, and $\text{dist}_i$ denotes the length of the $i^{th}$ line segment in Equation (3.10).

$$\text{dist}_i = \sqrt{(x_i - x'_i)^2 + (y_i - y'_i)^2}$$  \(3.10\)

t and $t'_i$ are the type of the two vertices. $\text{ang}_i$ is the direction of the $i^{th}$ line segment as Equation (3.11).

$$\text{ang}_i = \tan^{-1}\left(\frac{y'_i - y_i}{x'_i - x_i}\right)$$  \(3.11\)

and the range of $\text{ang}$ is $[0, 2\pi]$. Iridge num denotes the ridge numbers between two vertices. $i_{\text{cn}}$ is the connection numbers of the $i^{th}$ minutiae. The difference of parameter dist is smaller than a threshold, in the Equation (3.12)
Then the two line segments is defined to be equal, as given in the Equation (3.13).

\[
\text{abs}(\text{dist}_i^p - \text{dist}_i^q) \leq T_s
\]  

(3.12)

\[
\text{segment} P[i] = \text{segment} Q[j]
\]  

(3.13)

Figure 3.19  The line segment map of the two Fingerprints

The line segment map discerns the minutiae points with different samples. The segment mapped fingerprint samples lead to matching score. The matching process presented in the subsequent session.

3.7.2  Matching Score

The segment P[i] is matching to segment Q[j], that is, the match of these two line segments. The matching minutiae is divided into 4 Stages:

Stage.1. The minutiae belong to the structure which must have at least three line segments connected to each other, like area A in the Figure 3.19.(a), and each line segment meets the definition 2.
Stage.2. The minutiae belong to the structure which has two line segments connected to each other, like area B in the Figure 3.19.(a), and each line segment meets definition 2 too.

Stage.3 Minutiae are the two vertices of a matching line segment, like the area C in Figure 3.19. (a).

Stage.4. only the minutiae is matching, but the line segment with this minutiae is not matching.

Figure 3.20  Segment P[i ] and segment Q[ j] of two different Fingerprints

The value of matching score is calculated using Equation (3.14)

\[ mscore = \frac{\left(3N_4 + 2N_3 + 1.5N_2 + N_1\right)}{\max\{M, N\}} \]  

(3.14)

Where Ni denotes the number of matching minutiae in classification (i=1, 2, 3, 4). The matching score can be used to identify the imposter prints in recognition. Based on the number of scores matched the mscore will be used.
3.8 EXPERIMENTAL RESULTS

This session discusses the equipment and experimental procedures used to authenticate a person based on the fingerprint modality. The system is implemented using MATLAB 2009 in Windows 8 Operating System with 4 GB RAM.

The experimental results are compared with various existing work like point based recognition, image based recognition and graph based recognition. Segmentation elucidates better accuracy with FAR, FRR and high recognition rate. Every method just pose features extraction results with identity label. The parameter comparison between the existing methods and proposed segmentation based method is shown in the Figure 3.21, 3.22 and 3.23. Table shows the comparison of the proposed segmentation and existing algorithm with parameter results.

![False Acceptance Rate](image)

Figure 3.21 False Acceptance Rate
Figure 3.22  False Rejection Rate

Figure 3.23  Recognition Rate Comparison
Table 3.1 FAR, FRR and Recognition Rate Comparisons

<table>
<thead>
<tr>
<th>Methods</th>
<th>FAR</th>
<th>FRR</th>
<th>Recognition Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segmentation (proposed)</td>
<td>0.1684</td>
<td>1.7264</td>
<td>81%</td>
</tr>
<tr>
<td>point pattern</td>
<td>0.1841</td>
<td>1.6210</td>
<td>69%</td>
</tr>
<tr>
<td>image based</td>
<td>0.1919</td>
<td>1.1998</td>
<td>61%</td>
</tr>
<tr>
<td>Graph based</td>
<td>0.1794</td>
<td>1.2818</td>
<td>66%</td>
</tr>
</tbody>
</table>

3.8.1 Experimental Setup

This session discusses the equipment and experimental procedures used to authenticate a person based on the modality face. The system is implemented using MATLAB 2009 in Windows 8 Operating System with 4 GB RAM.

CASIA fingerprint images are used for fingerprint recognition. 200 users images are used for training and testing. Total number of training and testing done by using 400 fingerprint images from CASIA databases. CASIA Fingerprint Image Database Version 5.0. The fingerprint images of CASIA-FingerprintV5 were captured using URU4000 fingerprint sensor in one session. The volunteers were asked to rotate their fingers with various levels of pressure to generate significant intra-class variations. All fingerprint images are 8 bit gray-level BMP files and the image resolution is 328*356.

3.8.2 Parameter Estimation

The Segmentation Based Enhanced Fingerprint Recognition (SBEFPR) uniqueness model has several parameters, namely, r, l, w, A, m, n,
and q. The value of l further depends on $\theta_0$. The values of r, l, and w are estimated in this section for a given sensor resolution. The value of $r_0$ should be determined to account for the variation in the different impressions of the same finger is shown in the Figure 3.24

![Figure 3.24](image)

**Figure 3.24** Automatic matching of minutiae

The value of r was found to be 15 pixels for fingerprint images scanned at the resolution of 500 dpi. The value for $\theta_0$ was found to be $\theta = 22.5^\circ$. The distribution is $P(\min(|\theta' - \theta|, 360^\circ - |\theta' - \theta|) \leq 22.5^\circ) = 0.267$, i.e., $l = 0.267$. Under the assumption that minutiae directions are uniformly distributed and the directions for the minutiae that match in their location are independent, which obtain $l = (2 \times 22.5^\circ) / 360^\circ = 0.125$. For FP sensors with the resolution of 500 dpi, the ridge period is converted to $\sim 9.1$ pix./ridge $\Rightarrow w \sim 9.1$. 
Figure 3.25 Area of overlap between the two fingerprint that are matched based on the bounding boxes of the minutiae features for (a) MSU DBI database; (b) MSU VERIDICOM database.

It is observed from Figure 3.25 overlapping area between the two fingerprints based on the bounding boxes of the minutiae features in MSU_DBi database 0.28000 pixels but in the MSU_VERIDICOM database have 0.065 probability in 1000 pixels.

Figure 3.26 Comparison of experimental and theoretical probabilities for the number of matching minutiae. (a) MSU DBI database; (b) MSU VERIDICOM database.
The probability of number of matching minutiae in are calculated using MSU DBI database and compared with other methods. In empirical method the probability starts with 0.1 and it reaches the maximum value of 0.3. The theoretical method starts with the probability of 0 and reaches the peak of 0.15. In MSU_VERIDICOM database empirical reaches the value of 0.4 and theoretical reaches the probability of 0.23. It is clearly shows that using MSU_VERIDICOM database images matching minutiae values is higher than another databases.

3.9 SUMMARY

Thus this chapter presented the proposed Global Variational algorithm for segmentation based fingerprint matching. Global Variational representation, its methodology, and the various parameters adopted for segmentation matching. The impact of various parameters on segmentation based fingerprint matching was analysed in depth. A detailed comparison of FAR, FRR and EER has been calculated and concluded that segmentation based fingerprint matching process performs better.

The experimental results shows that the proposed segmentation based fingerprint recognition has the less False Acceptance Rate of 0.1684, False Rejection Rate of 1.7264 and mostly importantly the success rate of 81% when compared to other algorithms. The main contributions of this work has to enhance the accept rate of the fingerprint biometrics authentication for improve the matching process and strong feature extraction with more accuracy. Personal identification is tracked by another biometric trait, the Iris recognition.