FOURIER TRANSFORM BASED FINGERPRINT IMAGE PROCESSING FOR ACCURATE MINUTIAE DETECTION AND FAST COMPUTATION

[1]PROF. NEERAJ BHARGAVA, [2]DR. RITU BHARGAVA, [3]SURBHI SINGH, [4]PRAMOD SINGH RATHORE, [5]SURESH CHOUDHARY, [6]APOORVA BHOWMICK, [7]DR. ANKUR GOSWAMI

[1]Department of Computer Science, School of Engineering & System Sciences, MDS University,Ajmer, India
[2]Department of Computer Science, Sophia Girls' College (Autonomous), Ajmer, India
[3]Aryabhatta College of Engineering and Research Centre, Ajmer, India
[4]Department of Computer and Communication Engineering, Manipal University Jaipur, India
[5]Research Scholar, MDS University,Ajmer, India
[6]Information Systems Officer, United Nations Secretariate, New Yourk, NY, 10017, USA,
Baghdad, Iraq
[7]Dept. Of Computer Applications, Govt.
Engineering College Bikaner, RIICO Karni Industrial Area, Bikaner
E-mail: [1]profneerajbhargava@gmail.com, [2]drritubhargava92@gmail.com, [3]ssurbhi91@gmail.com,
[4]pramodrathore88@gmail.com, [5]er.choudharysuresh@gmail.com, [6]apporva.bk@gmail.com,
[7]drankursdgoswami@gmail.com

Abstract - This paper presents a comprehensive review of fingerprint image enhancement by applying a multi-stage processing pipeline, with the focus being on the detection of key minutiae features, namely bifurcation and termination points. Contrast Limited Adaptive Histogram Equalization, normalization, Fourier enhancement, binarization, thinning, and gradient filtering make up the pipeline. The influence of each stage on the number of minutiae detected is assessed with reference to a set of fingerprint images. Results show that although the different stages of processing in general, particularly CLAHE and Fourier enhancement, considerably improve the contrast and clarity, they also add subtle variations into minutiae detection, most of which are related to noise reduction or ridge structure enhancement. On the whole, final processed images balance enhanced details with minimal noise, which explains why the counts of minutiae remain very close to those detected in the original images. This consistency is important for the preservation of accuracy and reliability in the fingerprint recognition system. Results underline the role each stage can play during the enhancement process to refine features of fingerprints, which will be helpful in developing robust biometric identification technologies.

Keywords - Image, Fingerprints, Pixel, Gabor filter, Fourier Transform

I. INTRODUCTION

Fingerprint recognition systems are among the oldest and most trustworthy forms of biometric identification; they have found a wide range of applications because of their non- intrusive, precise, and unique nature. Their core is the extraction of minutiae points, basically bifurcations and terminations, functioning as the main features for fingerprint matching. However, this process of extracting minutiae depends much on the quality of the fingerprint images [1], [2]. Noisy, poorly contrasted, or distorted images significantly deteriorate the detection of these important features and hence attenuate the overall performance of the recognition systems [3],[4]. Some methods to enhance fingerprint image quality and minutiae detection have been proposed in this direction to enhance accuracy. Of these, Fourier transform-based methods have gained attention due to their capability to bring out ridge patterns of fingerprints by selective enhancement of some frequency components [5],[6]. The Fourier transform is especially good at bringing out periodicity in ridge structures, and this significantly increases the detectability of bifurcation and termination points. This may then be combined with other preprocessing techniques, including normalization, noise reduction, adaptive histogram equalization, binarization, and directional filtering, to prepare the image for Fourier-based enhancement and features extraction [7],[8]. The paper proposes a comprehensive framework of fingerprint image enhancement in which Fourier transform- based enhancement is combined with a number of preprocessing steps to work out the optimal minutiae points detection. First, normalization and noise reduction are applied to standardize the image intensity and remove unwanted noise [9], [10]. Next, gradient computation, ridge frequency and coherence calculation, and directional filtering are performed in sequence to further refine the image. First, ridge patterns are enhanced using the Fourier transform, followed by various steps in image enhancement at the frontend, such as CLAHE and Gabor filtering, followed by region masking to bring out a clearer and more detailed image [11], [12]. This research focuses not only on image enhancement itself but also on examining the computational efficiency for each processing step. In the study, the time taken for each enhancement step is measured because in real-life

biometric systems, real-time performance is important [13], [14]. This will show the compromise that a real-world application would require between improvements in image quality and processing speed.

These results lend weight to the development of more robust fingerprint recognition systems by showing that Fourier transform-based enhancement is a very effective way to couple it with other preprocessing techniques. This approach enhances reliability and accuracy in fingerprint-based identification systems by providing better bifurcation and termination point detection [15], [16]. Further, the analysis of the processing times reports some practical guidelines related to the application of these techniques in those environments where both accuracy and efficiency are important [17].

II. LITEARTURE SURVEY

Fingerprints consist of distinct ridge and valley patterns on a fingertip, ideal for personal identification due to their uniqueness and permanence [18]. These patterns develop in the womb and remain constant throughout a person's lifetime. The unique characteristics of fingerprints can be categorized into types, patterns, minutiae points, and secondary attributes.

2.1 Types of Fingerprints

(a) Live Scan Fingerprint: Captured electronically using specialized scanners.
(b) Latent Fingerprint: Invisible traces left by natural oils and sweat on touched objects.
(c) Patent Fingerprints: Visible prints left by fingertips that have come into contact with liquids like blood or ink.

2.1.(i) Patterns of Fingerprint: Fingerprint patterns can be classified into three major categories:

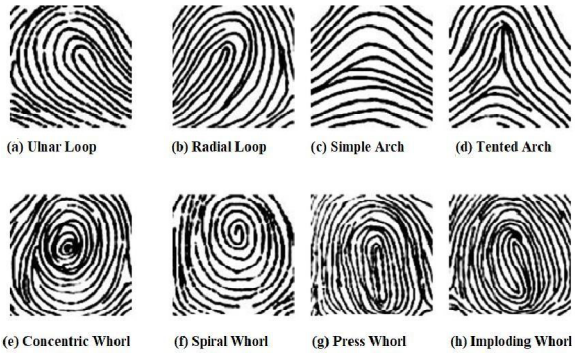


Figure 1: Patterns of fingerprint

2.1.(ii) Minutiae Points

Minutiae points are classified into ridge endings and ridge bifurcations. These points are critical for fingerprint analysis and identification [18].

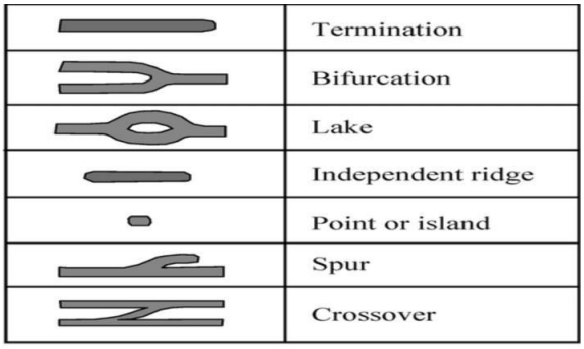


Table-Common Minutiae Points of Fingerprint

Table 1: Minutiae Points

III. RELATED WORK

Proposed method integrates two filters to enhance fingerprint images. The initial stage, normalization, adjusts the image's grayscale intensity values to fall within a predefined range, enhancing visual contrast and providing a greater range of intensity levels for further processing (Figure 2). This step is crucial as uneven pixel distribution and noise during scanning can compromise the quality of fingerprint details.

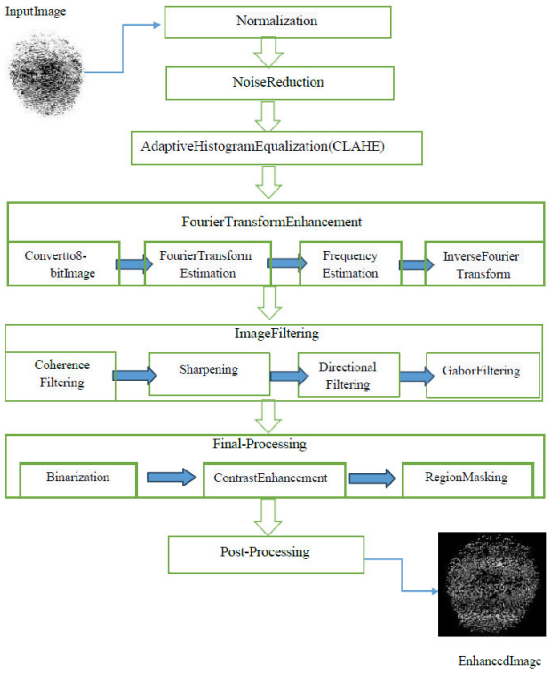


Figure 2: Flow Diagram of Fingerprint Enhancement

3.1 Normalisation algorithm for finger skin profiles

Normalization techniques, like the ones introduced by Hong et al., decrease the range of grayscale values between ridges and valleys. Approach visualizes the fingerprint image as a three-dimensional structure, utilizing positional coordinates on the x-y plane and pixel grayscale values on the z-plane. This approach ensures that ridge information is preserved, which is vital for minutiae-based fingerprint identification systems (Figure 3).

The steps for fingerprint image normalization are as follows:

1. Calculate the mean intensity of the original image.
2. Calculate the variance of the original image.
3. Normalize the intensity values to obtain the normalized image.

Even in cases where skin profiles are uneven, Technique ensures that ridge information is preserved, which is crucial for retrieving minute details.

- $I_n(x, y)$ be the intensity of the pixel at position (x, y) in the original image.
- (x, y)in the normalized image.
- $M_i$be the mean intensity of the original image.
- V be the variance of the original image.
- M' be the desired mean intensity.
- V' be the desired variance.

The steps for fingerprint image normalization are as follows:

(i) Calculate the mean intensity M_i of the original image:

Where$k$isthesizeofthe kernel.

$$M_i=\frac{1}{N}\sum_{x=1}^{W}\sum_{y=1}^{H} I(x,y)$$

whereWandHarethewidthandheightoftheimage,and N=W×Histhetotalnumberofpixels.

(i) Calculatethevariance$V$oftheoriginalimage:

$$V=\frac{1}{N}\sum_{x=1}^{W}\sum_{y=1}^{H} (I(x,y)-M_i)^2$$

Normalizetheintensityvaluestoobtain$I_n(x,y)$

$$I(x,y)=M'+\sqrt{\frac{V'}{V}}(I(x,y)-)$$

n i



Figure 3: Fingerprint Normalized Image

3.2 Noise reduction using Gaussian Filtering

Image denoising is a very essential step in the process of image processing to ensure clear images. As a linear smoothing filter, a Gaussian filter is efficiently used for reducing noise by applying a Gaussian function on pixels in the neighbourhood of interest. The weighting for the central pixels would be more than that at edges, which in turn averages out the pixel values to obtain a smoothened image. It gives more weight to pixels lying near the centre of the kernel and, therefore, smoothies high-frequency noise while preserving edges and fine details (Figure 4).

Gaussian Filter Formula

Given an image I, the Gaussian filter works by convolving the image with a Gaussian kernel. The Gaussian kernel G is defined as:

$$G(x,y)=\frac{1}{2\pi\sigma^2}exp^{-(\frac{(x^2-y^2)}{2\sigma^2})}$$

Where $\boldsymbol{\sigma}$ is the standard deviation of the Gaussi an distribution, and x and y are the distances from the centre of the kernel in the horizontal and vertical directions, respectively.

The filtered image $I_{filtered}$ is obtained by convolving the image I with the Gaussian kernel G:

$$I_{filtered}(i,j)=\sum_{x=-k}^{k}\sum_{y=-k}^{k} I(i+x,i+y)G(x,y)$$

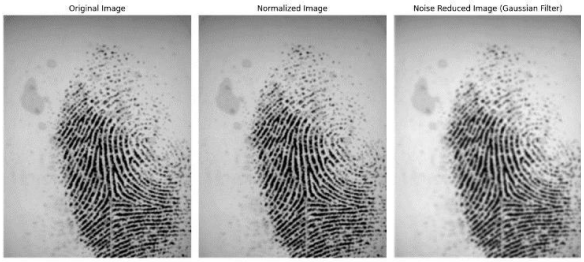


Figure4:Fingerprint Noise Reduced Image

3.3 Histogram Equalization

Histogram equalization This is one of the methods used in enhancing the contrast of an image by the redistribution of the intensity values to contrast enrichment. It stretches out the most frequent intensity values for better visibility of features across the image. It improves image detail and balance by using the cumulative distribution function, which maps original intensities to some new and uniform distribution. (Figure 5 to 8).

Mathematical Formula

Given an image$I$ with L possible intensity levels $0to$ $(L-1)$, the steps to perform histogram equalization are as follows:

1. **ComputetheHistogram:** TheHistogramH(M_k) representsthenumberofpixelswithintensityM_k.

$$II(M_k)=\sum_{x=1}^{W}\sum_{y=1}^{H}\delta(I(x,y)-M)_k$$

Where:

- δistheKroneckerdeltafunction.
- I(x,y)istheintensityofthepixelatposition (x,y).
- WandHarethedimensionsoftheimage.

2. **ComputetheProbability Distribution:** Normalize the histogram to get the probability distribution of each intensity level.

$$P(M)_k=\frac{H(M_k)}{WH}$$

3. **Compute theCumulativeDistributionFunction (CDF):**

$$CDF(M_k)=\sum_{y=0}^{k}P(M_y)$$

4. **MaptheIntensityLevels:**

Map the original intensity levels to the new levels using the CDF

$$S_k=(L-1)CDF(M_k)$$

WhereS_k is the new intensity level corresponding to the original level M_k.

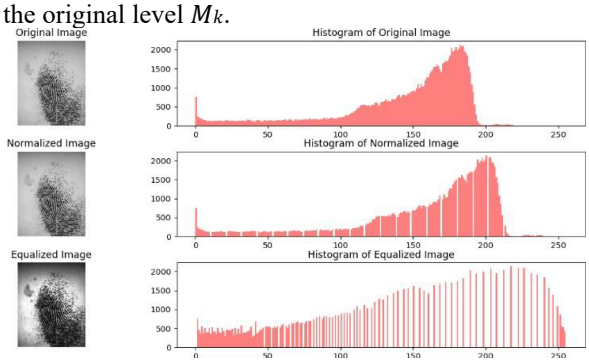


Figure5: Fingerprint Equalized Image

5. Adaptive Histogram Equalization

Adaptive histogram equalization will ensure better image contrast using only local contrast adjustments instead of the global ones. Contrast Limited Adaptive Histogram Equalization, CLAHE, is a variant of AHE that aims to avoid noise over-amplification. Here is the full script now having adaptive histogram equalization using CLAHE, normalization, and visualization:

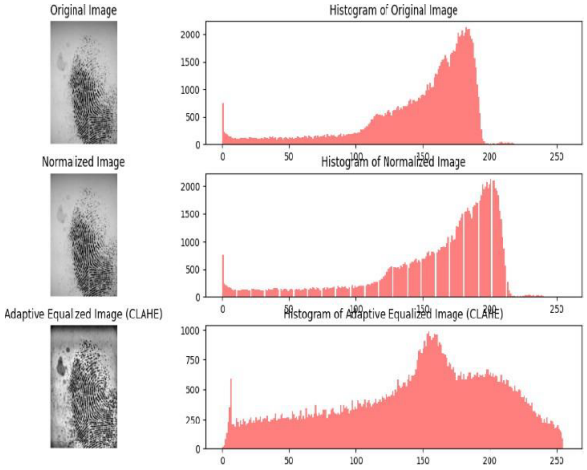


Figure 6: Fingerprint Adaptive Equalized Image (CLAHE)

6. Adaptive Histogram Equalization Conversion to 8-bit

After applying CLAHE, the resulting image is often in floating-point format with values ranging from 0 to 1. To convert it to an 8-bit image, we scale the values to the 0 to 255 range:

$$I_{8-bit}=(_{oat}\times 255)\text{astype(np.uint8)}$$

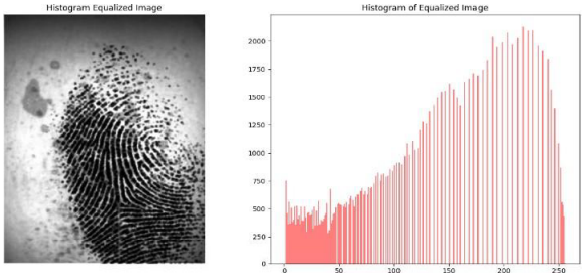


Figure 7: Adaptive Histogram Equalization Conversion to 8-bit

7. Contrast Enhancement:

Basically, contrast enhancement in images makes features more visible by increasing the difference between light and dark areas. Adaptive histogram equalization, AHE, and its variant, CLAHE, are generally strong methods for local contrast enhancement of digital images; this paper includes general contrast enhancement to improve the image quality. The enhance contrast function enhances the contrast of the 8-bit adaptive equalized image using the Image Enhance. Contrast class from the PIL library.

$$I_{enhanced}(\text{i,j})=(i,j)+\beta$$

whereα is the contrast factor and β is the brightness adjustment. For simplicity, we use a PIL function.

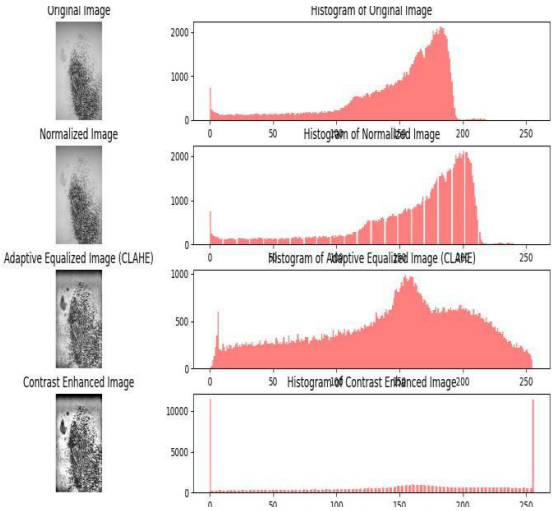


Figure 8: Fingerprint Contrast Enhanced Image

3.4 Thinning of Fingerprint Image

Thinning reduces ridges in a binary image to lines of a single pixel width, simplifying the image for subsequent processes like minutiae extraction. The Zhang-Suen algorithm is used for this step (Figure 9).

Initialization:

Start with abinary image $I_B$where ridges are represented by black pixels (1) and valleys by white pixels (0).

- **FirstSub-iteration:**

Foreachpixel(x,y)intheimage,apixelP(x,y)isremoved ifitsatisfiesthefollowing conditions:

1. ThepixelP(x,y)isablackpixel.
2. $(2 \leq N(P(x,y)) \leq 6$ where $NP(x,y))$ isthe numberofblackneighborsof P(x,y).
3. $S(P(x,y))=1$,where$S(P(x,y))$isthenumberof 0 to1transitionsintheorderedsequenceof $P_2,P_3.....P_8,P_9,P_2$.
4. AtleastoneofP_2,P_4,P_6isawhitepixel(0).
5. AtleastoneofP_4,P_6,P_8isawhitepixel(0).

- **SecondSub-iteration:**

Foreachpixel(x,y)intheimage,apixelP(x,y) isremoved if it satisfies the following conditions:

1. ThepixelP(x,y)isablackpixel.
2. $(2 \leq N_1(P(x,y)) \leq 6$ where$N_1P(x,y))$isthe number of black neighbors of P(x, y).
3. $S(P(x,y))=1$,where$S(P(x,y))$isthenumberof 0 to1transitionsintheorderedsequenceof $P_2,P_3.....P_8,P_9,P_2$.
4. AtleastoneofP_2,P_4,P_6isawhitepixel(0).
5. AtleastoneofP_4,P_6,P_8isawhitepixel(0).

- **IterateUntilStable:**

Repeattheabovetwosub-iterationsuntilnomorepixelscan be removed, meaning the image does not change between iterations.

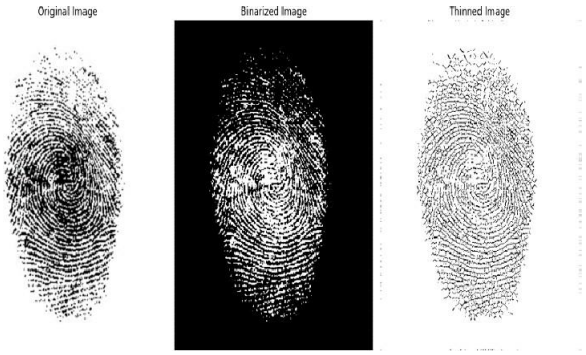


Figure 9: Thinned Image

3.5 Log-Gabor filter

The Log-Gabor filter, employed in texture analysis and pattern recognition, excels at enhancing images containing diverse frequency ranges. It enhances the contrast between ridges and valleys while maintaining the integrity of ridge structures (see Figure 10).

The2DLog-Gaborfilterisdescribedinthespatialdomainas follows:

$$g(x,y;f_o,\theta,\sigma_f,\sigma_\theta)=\frac{1}{2\pi\sigma_f\sigma_\theta}\exp\left(-\frac{\log^2\left(\frac{f(x,y)}{f_0}\right)}{2\log^2(\sigma_f)}-\frac{\theta^2}{2\sigma_\theta^2}\right)$$

where:

- (x,y)arethespatialcoordinates.
- $f(x,y)$isthelocalspatialfrequencyat position (x,y).
- $f_0$isthecentralfrequencyparameteroftheLog-Gabor filter.
- θistheorientationoftheLog-Gaborfilter.
- $\sigma_f$controlsthebandwidthofthefilterinthe frequency domain.
- $\sigma_\theta$controlsthebandwidthofthefilterinthe orientation domain.

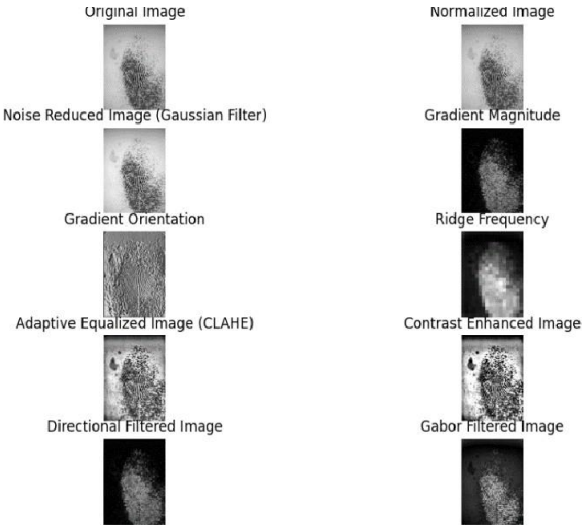


Figure 10: Log-Gabor Filtered Image

3.6 Estimation of Ridge Orientation

Ridge orientation is estimated using gradient magnitude and orientation analysis.

This involves calculating the gradient magnitude and orientation at each pixel and constructing an orientation histogram to determine the dominant ridge orientation across the fingerprint image (see Figure 11&12).

ComputeGradientMagnitudeandOrientation:

Calculatethegradientmagnitude$|\nabla I|$andorientation $\vartheta(x,y)$)ateachpixel (x,y).∇

$$|\nabla I| = \sqrt{I_x^2 + I_y^2}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{I_y}{I_x}\right)$$

where$I_x$and$I_y$arethepartialderivativesoftheimage intensity $I(x, y)$ with respect to x and y respectively.

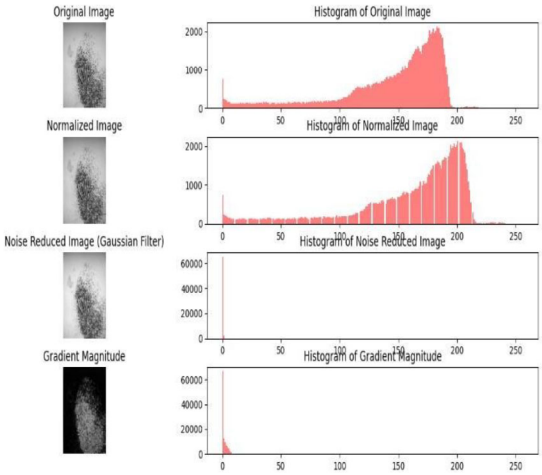


Figure11:FingerprintGradientMagnitudeImage

- **Accumulate Orientation Histogram:**

Construct an orientation histogram to determine the dominant ridge orientation across the fingerprint image.

This histogram counts gradient orientations within localized regions, typically using orientation bins. Find Peak(s) in Orientation

Histogram: Determine the peaks in the orientation histogram, which indicate the primary ridge orientations in the fingerprint.

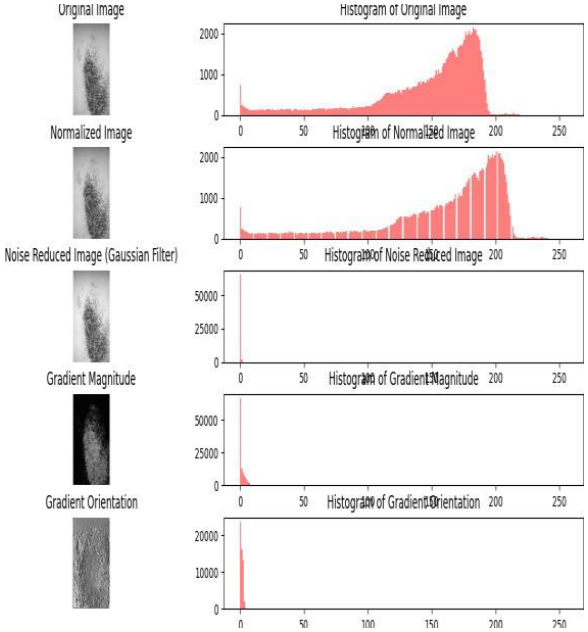


Figure 12: Fingerprint Gradient Oriented Image

3.7 Ridge Frequency:

Ridge frequency, the distance between successive fingerprint ridges, is estimated using local ridge period analysis and local power spectrum analysis (Figure 13).

LocalRidgePeriodAnalysis:

To estimate the ridge frequency (x,y) , apply Fourier analysisorotherfrequencydomainmethodswithinparticular portions of the fingerprint image.

AnalysisofLocalPowerSpectrum:

Compute the local power spectrum(f_1,f_2),using Fourier transformtechniques,where$f_1$and$f_2$arespatialfrequencies.

$$(f_1,f_2) = |(f_1,f_2)|^2$$

Identifythepeakfrequencycomponentsinthepower spectrum corresponding to the ridge frequency.

Peak Detection:

Detectthepeakfrequencyinthepowerspectrumtoestimate the local ridge frequency (x, y)

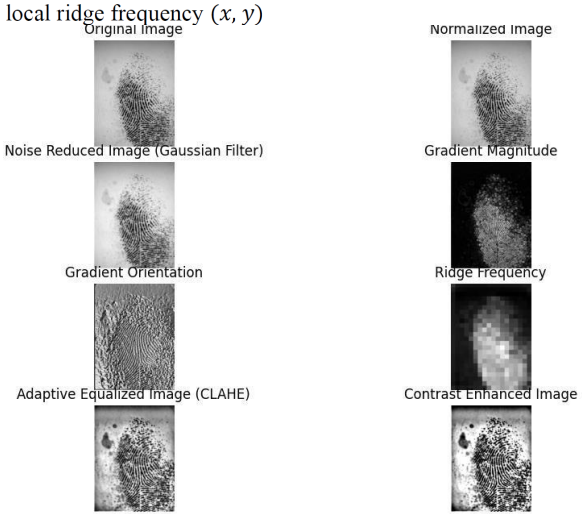


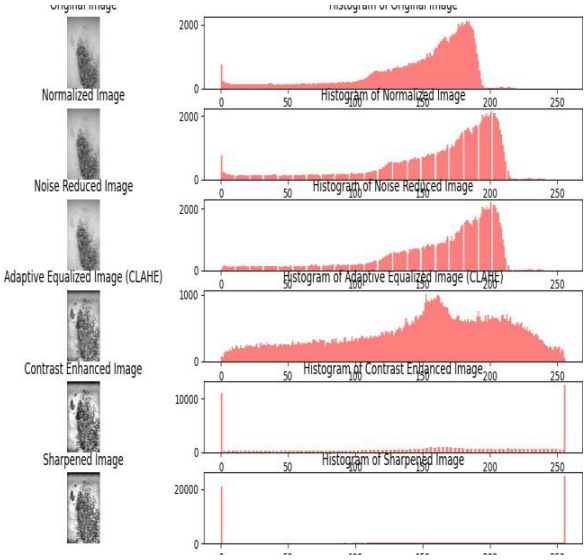
Figure 13: Ridge Oriented Image

3.8 Sharpening

Sharpening increases the clarity of an image by highlighting edges and fine details through insisting on differences in intensity. The sharpening kernel, for example, convolves the image. What happens in these processes is that contrast gets increased around edges, which are emphasized and sharpness improved. Sharpening forms the basis of many image sharpening operations, aimed at bringing clarity and details into images (Figure 14).

The sharpened image is obtained by convolving the image I with the sharpening kernel

$$I_{sharpened}(i,j) = \sum_{x=-k}^{k} \sum_{y=-k}^{k} I(i+x, j+y) K(x,y)$$



3.9 Coherence filter

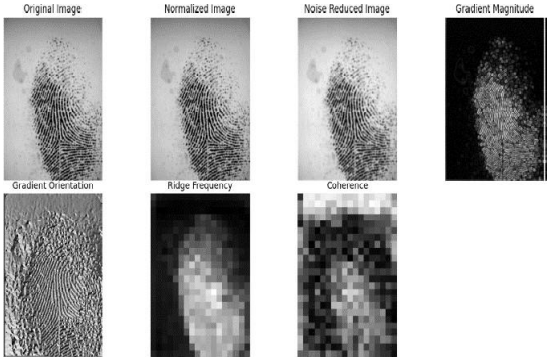
The coherence filter is a very vital technique in fingerprint image enhancement. It deals with the measurement of gradient-orientation coherence within every localized region in an image. This filter is used to differentiate a well-defined ridge pattern from noise, making fingerprint features very clear for analysis. (Figure 15).

Coherence Calculation:

It involves the summing of cosine and sine components in every local block, after which coherence is computed.

$$C = \frac{\sqrt{(\sum \cos(2\theta))^2 + (\sum \sin(2\theta))^2}}{N}$$

Where N is the number of pixels in the block. Coherence values range from 0 (no coherence) to 1 (perfect coherence).



3.10 Region Mask

One may design a region mask of a fingerprint image by locating some regions of interest in an image mathematically.

This kind of processing will turn out to be beneficial for several applications related to feature extraction, matching, and analysis (Figure 16). Some of the steps and formulas used are as follows:

Preprocess the Image:

- Convert the image to grayscale.
- Apply Gaussian blur to reduce noise.
- Enhance the image contrast.

1. **GrayscaleConversion:**
   ConvertanRGBimageto grayscale

$$I_{gray} = (0.2989R + 0.5870G + 0.1140B)$$

2. **GaussianBlur:**
   Gaussianfiltertosmooththeimage

$$G(x,y) = \frac{1}{2\pi\sigma^2} exp^{-\left(\frac{(x^2-y^2)}{2\sigma^2}\right)}$$

ConvolvetheimageIgraywiththeGaussiankernelG:

$$I_{blurred}(x,y) = a_y \times G(x,y)$$

3. **Otsu'sThresholding:**
   Findthethresholdthatminimizestheintra-class variance:

$$\sigma^2(t) = \omega(t)\sigma^2(t) + \omega(t)\sigma^2(t)$$

o 1 1 2 2

Where:

- ω_i istheprobabilityofclass i
- σ_i isthevarianceofclass i

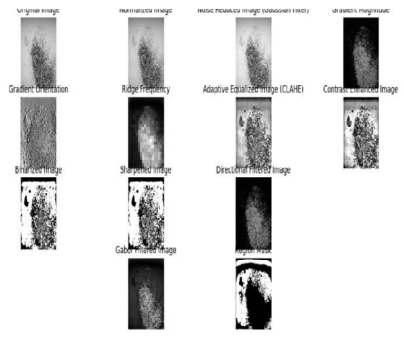


Figure16RegionMask Image

3.11 Fourier Transform

Basically, fingerprint enhancement using the Fourier Transform involves image conversion from the spatial domain to the frequency domain, application of filters for the enhancement of some features, and then a reverse conversion to the spatial domain. In the approach, ridge structures can be made more visible by isolating and enhancing only those specific frequency components that correspond structurally (Figure 17).

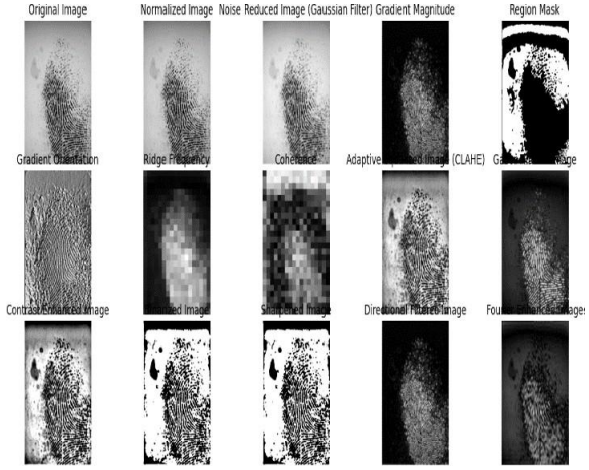


Figure 17 Fourier Enhanced Image

Key Concepts

1. FT: It is a process for changing an image from the spatial domain to the frequency domain.
2. Inverse Fourier Transform (IFT): Converts the image back from the frequency domain to the spatial domain.
3. Band-pass filtering: This will isolate and enhance the specific ridge structure frequency components.

1. FourierTransform

The2DDiscreteFourierTransform(DFT)ofan image $(x,y)$is givenby:

$$F(u,v)=\sum_{x=0}^{M-1}\sum_{=0}^{N-1} I(x,y)e^{-2j\pi(\frac{ux}{M}+\frac{vy}{N})}$$

Where:

- $I(x,y)$istheintensityofthepixelatcoordinates (x,y)
- $F(u,v)$isthefrequencycomponentatcoordinates (u,v)
- $M$and$N$arethedimensionsoftheimage
- $j$istheimaginaryunit

2. FrequencyDomainFiltering

Design a bandpass filter to isolate the frequencies correspondingtotheridgepatterns.AGaussianbandpass filter can be used:

$$H(u,v)=e^{-(\frac{(D(u,v)-D_0)}{2\sigma^2})}$$

where:

- $D(u,v)=\sqrt{(u-M/2)^2+(v-N/2)^2}$ is the distance from the center of the frequency domain
- D_0isthecenterfrequencyofthebandpassfilter
- σcontrolsthebandwidthofthefilter

3. InverseFourierTransform

The Inverse 2D Discrete Fourier Transform (IDFT) is usedtoconvertthefilteredfrequencycomponentsbackto the spatial domain:

$$I_{filtered}(x,y)=\frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}F(u,v)^{2j\pi(\frac{ux}{M}+\frac{vy}{N})}$$

3.8 Binarization

Binarization converts a grayscale image into a binary image, simplifying the identification of ridge structures. Otsu's thresholding method is used for this process (Figure 18).

Convertthegrayscale I imageintoabinaryimage I_B using the optimal threshold T^*.

$$I_B(x,y)=\begin{cases}1 & if\ I(x,y)\geq T^* \\ 0, & if\ I(x,y)<T^*\end{cases}$$

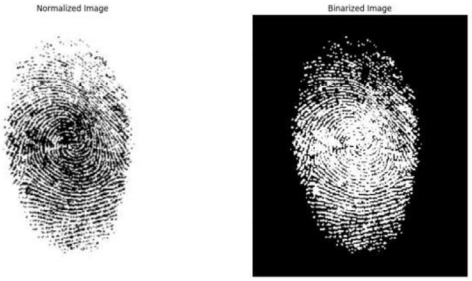


Figure18:BinarizedImage

3.9.(i) Minutiae Detection Process:

Minutiae detection involves analysing the neighbourhood of each pixel in the thinned image to identify ridge endings and bifurcations. This step is crucial for fingerprint analysis (Figure1 9).

(a) Thinning: Ensure the fingerprint ridges are reduced to a single pixel width. This simplifies the structure of the fingerprint, making it easier to detect minutiae points (figure 9).
(b) Neighborhood Analysis: For each pixel in the thinned image, analysis the 3x3 neighborhood to determine if it is a minutia point.

3.9.(ii) Neighborhood Analysis Formula: For a given pixel at coordinates (x, y) let P(i, j) represent its

binary value (1 for ridge, 0 for valley). Consider the 3x3 neighborhood around P(i, j).

3.9.(iii) Minutiae Classification:

Ridge Ending Detection: A ridge ending is identified if the central pixel P(x, y) is 1 (ridge) and the sum of its 8 neighbours equals 1.

$$\text{Formula:} \sum_{z=1}^{8} P_z = 3$$

Transition Count Method:

An alternative method to confirm minutiae involves counting the number of transitions from 0 to 1 in the neighborhood. For each pixel P(x, y), compute the number of transitions in its neighbourhood in a circular manner:

$$N = P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_5 \rightarrow P_6 \rightarrow P_7 \rightarrow P_8 \rightarrow P_1 \rightarrow P_2$$

LetT(P)bethenumberoftransitions(0to1)inthesequence:

- **RidgeEnding**:T(P)=1
- **RidgeBifurcation**:T(P)=3

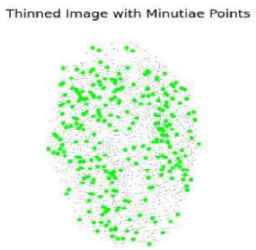
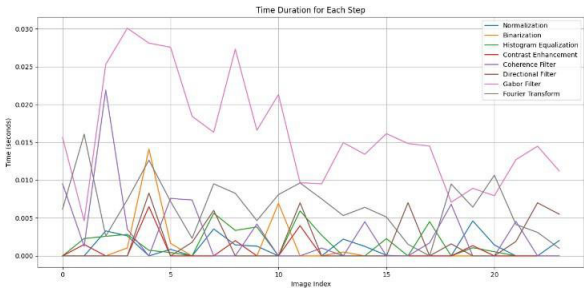


Figure 19: Thinned Image with Minutiae point

3.10 Time Cycle Detection:

Time cycle detection refers to the processing sequence or computational time required for each step in the fingerprint enhancement process. This involves evaluating the time taken for normalization, binarization, thinning, and minutiae detection (Graph 1).

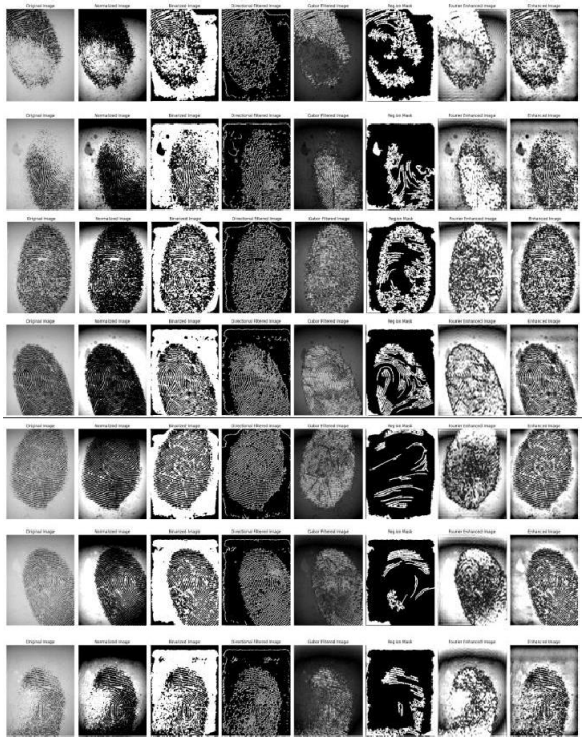
- Normalization: Typically, fast; involves calculating mean and standard deviation over the entire image.
- Binarization: Relatively fast; requires a single thresholding operation.
- Thinning: More computationally intensive; involves iterative operations over the image until convergence.
- Minutiae Detection: Depends on the image size and complexity; involves analysing each pixel and its neighbourhood.



Graph 1: Experimental Result Graph

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed method was implemented in Visual Studio Code using Python, Matplotlib,Pandas,NumPy,andOpenCVlibrariesonthe FingerprintVerificationCompetition(2002) (FVC) fingerprint image dataset. The results demonstrate significant improvements in image clarity and minutiae detection after applying the enhancement techniques. The average number of bifurcation and termination points increased after normalization and binarization, indicating improved visibility and distinctness of these points. The experimental results show that the normalization techniqueenhancesthequalityoffingerprintimages,making it easier to identify minutiae points. The Log-Gabor filter significantly improves image quality by enhancing contrast while preserving ridge structures. The combination of filters producesafinalenhancedimagewithbetterclarityanddetail (Figure 20, Graph 2-3, Table 2-3).



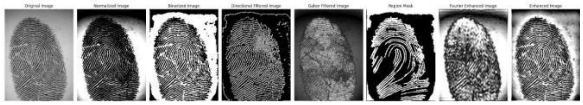
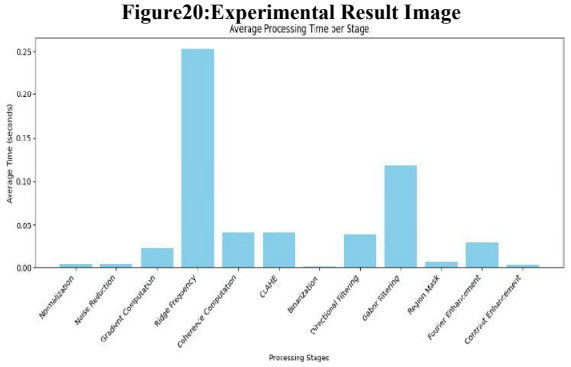


Figure20:Experimental Result Image

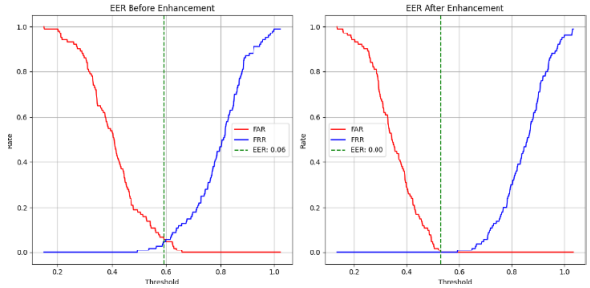


Graph 2: Experimental Result Graph

Average time for the processing stages has been analysed in the bar graph(Graph 2), and it is clear that the coherence computation is the stage with the highest average time compared to all other stages evaluated. It is well above all other stages by at least an order of magnitude, averaging around 0.25 seconds.

Other stages like Region Filtering and Directional Filtering also have moderate processing times, adding significantly to the total computational load but were less than that for coherence computation.

On the other hand, Normalization, Node Reduction, and Gradient Computation had very low processing times, indicating that these are computationally inexpensive operations in the pipeline. Contrast Enhancement and Fourier Enhancement also followed this pattern with negligible average times. This distribution indicates that there are major computational bottlenecks in the pipeline and hence optimization should primarily target coherence computation followed by the filtering stages. These findings are clearly pointing out what should be improved to enhance overall efficiency of the processing workflow.



Graph 3: Experimental Result Graph

To judge the performance of biometric systems, the False Acceptance Rate (FAR) and False Rejection Rate (FRR) are compared at the equal error rate (EER). In this paper, EER values are computed before and after enhancement processing.

The curve EER is at 0.06 at intersection. PerformanceThe FAR and FRRhaveglaringerrors,andthe biometric system's accuracy level is slightly well-balanced.

After Enhancement FAR vs. FRR Curve: The point of intersection falls at EER of 0.00, thereby indicating the presence of no measurable error in the balance between FAR and FRR. Performance The performance of the system shows a very high and optimised after enhancement and, therefore, highly accurate system.

EnhancementremovesEER,andsincebothFAR and FRR were perfect equilibrium, it indicated that these enhancement techniques applied significantly reduced EER from 0.06 to 0.00, proving effectiveness in enhancing the reliability of the biometric system under study. This work signifies how importance is attached to optimization in the performance of the biometric systems through enhancement techniques. The elimination of EER altogether emphasizes that post-enhancement, the biometric authentication system may become highly accurate and reliable. This is, in itself, a milestone in terms of system accuracy and robustness.

Image	Bifurcation Points (Original)	Termination Points (Original)	Bifurcation Points (CLAHE)	Termination Points (CLAHE)	Bifurcation Points (Normalized)	Termination Points (Normalized)	Bifurcation Points (Fourier Enhanced)	Termination Points (Fourier Enhanced)	Bifurcation Points(Binarized)	Termination Points (Binarized)	Bifurcation Points (Thinned)	Termination Points (Thinned)	BifurcationPoints (Gradient)	Termination Points (Gradient)	Bifurcation Points(Final)	Termination Points(Final)
101_1.tif	18	24	23	20	10	22	27	30	27	37	26	31	29	38	11	21
101_2.tif	29	33	15	39	11	21	11	39	10	27	13	37	25	34	20	37
101_3.tif	10	39	16	20	20	21	11	21	22	23	24	23	19	27	13	20
101_4.tif	27	34	26	22	17	25	10	29	20	23	24	24	23	21	28	23
101_5.tif	26	39	11	32	21	34	16	33	20	23	14	37	13	20	24	35
101_6.tif	23	32	29	36	14	33	28	27	24	31	14	21	15	39	15	28
101_7.tif	25	25	17	29	28	23	22	32	24	25	15	31	20	24	18	28
101_8.tif	27	28	20	28	27	34	26	21	22	23	22	33	21	20	17	24
102_1.tif	16	38	17	20	13	25	11	33	14	28	22	39	25	24	24	28
102_2.tif	25	25	10	23	17	20	21	37	26	30	16	28	12	33	20	29
102_3.tif	25	36	18	38	10	31	11	25	28	38	12	38	19	28	15	26
102_4.tif	29	36	17	27	23	28	16	24	19	39	11	27	19	36	21	38
102_5.tif	27	36	11	37	25	36	15	22	28	29	28	39	13	20	22	20
102_6.tif	22	34	29	30	26	31	20	24	12	26	18	29	13	26	25	29
102_7.tif	16	27	16	31	17	27	15	39	14	24	23	22	25	37	25	20
102_8.tif	11	34	21	27	14	33	18	39	10	28	18	21	16	39	25	37
103_1.tif	15	36	16	31	16	36	10	32	19	34	21	39	19	34	12	22
103_2.tif	12	25	13	33	29	26	29	20	18	35	20	24	23	28	21	39
103_3.tif	13	22	11	33	23	39	22	25	13	39	29	20	20	27	22	36
103_4.tif	22	27	28	21	27	27	24	35	15	34	14	36	13	32	13	33
103_5.tif	20	38	20	28	19	29	26	39	11	21	28	27	26	35	23	38
103_6.tif	29	30	27	31	16	39	25	35	18	21	24	39	29	26	28	29
103_7.tif	27	20	12	29	21	29	25	33	22	22	16	39	27	35	21	34
103_8.tif	24	20	12	30	23	37	13	39	17	20	17	20	28	24	24	39
finger.jpg	12	30	22	30	27	26	10	26	18	28	29	37	22	39	28	36
Average	**21.2**	**30.72**	**18.28**	**29**	**19.76**	**29.28**	**18.48**	**30.36**	**18.84**	**28.32**	**19.92**	**30.44**	**20.56**	**29.84**	**20.6**	**29.96**

Table:2 Timecycledetection,bifurcation point,and termination point of each stage

Image	Normalization	Noise Reduction	Gradient Computation	Ridge Frequency	Coherence Filtering	CLAHE Enhancement	Binarization	Directional Filtering	Gabor Enhancement	Region Mask Computation	Fourier Enhancement
101_1.tif	0.000000	0.000000	0.015678	0.078114	0.015625	0.062519	0.031194	0.015635	0.072529	0.012725	0.011302	0.006680
101_2.tif	0.000985	0.000000	0.014508	0.068636	0.026257	0.016458	0.001003	0.006084	0.062510	0.000000	0.015630	0.000000
101_3.tif	0.000000	0.000000	0.000000	0.078488	0.015636	0.015633	0.000000	0.015657	0.062509	0.000000	0.015641	0.000000
101_4.tif	0.015627	0.002108	0.001595	0.069884	0.016731	0.016711	0.000000	0.016610	0.088640	0.000000	0.015180	0.000000
101_5.tif	0.000000	0.000000	0.000000	0.079220	0.017403	0.017053	0.000000	0.033119	0.097927	0.006352	0.012248	0.000000
101_6.tif	0.000000	0.000000	0.016614	0.118888	0.051285	0.052772	0.003000	0.081059	0.105356	0.000000	0.022238	0.017013
101_7.tif	0.000000	0.001000	0.000000	0.098143	0.015613	0.015634	0.000000	0.015615	0.075810	0.000000	0.017310	0.001999
101_8.tif	0.002000	0.000078	0.000000	0.062558	0.015629	0.031190	0.000000	0.031253	0.102792	0.000000	0.015632	0.000000
102_1.tif	0.001007	0.002002	0.006998	0.064767	0.031295	0.031211	0.000000	0.031247	0.124995	0.000000	0.031253	0.000000
102_2.tif	0.000000	0.000000	0.017986	0.098762	0.002583	0.031220	0.000000	0.033122	0.047324	0.000000	0.015625	0.000000
102_3.tif	0.000000	0.000000	0.000000	0.062464	0.015628	0.031253	0.000000	0.015631	0.046866	0.000000	0.015639	0.000000
102_4.tif	0.015628	0.001886	0.001953	0.067888	0.002000	0.031323	0.000000	0.015625	0.071095	0.002005	0.015751	0.008709
102_5.tif	0.001993	0.004002	0.006204	0.082664	0.015730	0.015625	0.000000	0.015605	0.075902	0.002997	0.006996	0.000000
102_6.tif	0.000000	0.000000	0.000000	0.078076	0.015674	0.015578	0.000000	0.015629	0.081339	0.000000	0.015625	0.000000
102_7.tif	0.000000	0.015627	0.000000	0.113862	0.034900	0.028280	0.001000	0.047907	0.124371	0.000000	0.015628	0.015624
102_8.tif	0.000993	0.002999	0.006652	0.075568	0.015617	0.015627	0.000000	0.015630	0.046916	0.015621	0.000000	0.015580
103_1.tif	0.000000	0.010631	0.007014	0.058499	0.007004	0.015704	0.000000	0.015627	0.062498	0.000000	0.015616	0.000000
103_2.tif	0.000000	0.000000	0.000000	0.080224	0.014612	0.009093	0.000000	0.015692	0.062483	0.000000	0.015634	0.000000
103_3.tif	0.000000	0.000000	0.015655	0.052100	0.031350	0.016862	0.000000	0.003997	0.062576	0.000000	0.015622	0.000000
103_4.tif	0.001012	0.001533	0.002486	0.064985	0.016909	0.015429	0.000000	0.019124	0.060523	0.000000	0.015625	0.000000
103_5.tif	0.000000	0.000000	0.000000	0.062258	0.015626	0.015627	0.000000	0.015624	0.054970	0.002010	0.000000	0.000000
103_6.tif	0.000000	0.000000	0.000000	0.171870	0.031259	0.042558	0.001008	0.034170	0.136120	0.013799	0.043499	0.000000
103_7.tif	0.000000	0.000000	0.015626	0.156250	0.031248	0.016913	0.000000	0.024318	0.056403	0.015627	0.000000	0.000000
103_8.tif	0.000000	0.000000	0.015630	0.071848	0.015587	0.015671	0.000000	0.015629	0.046814	0.015625	0.000000	0.000000
finger.jpg	0.000000	0.000000	0.015635	0.191162	0.046842	0.064775	0.002003	0.050962	0.156236	0.000000	0.062504	0.000000
Average	**0.001570**	**0.001675**	**0.006409**	**0.088287**	**0.020722**	**0.025629**	**0.001568**	**0.024023**	**0.079420**	**0.003470**	**0.016408**	**0.002624**

Table: 3 Time cycle detection, bifurcation point, and termination point of each stage

Bifurcation Points (Original/Normalized): Average of 21.20 points before enhancement, increased to an average of 20.00 after normalization, and further to 18.82 after binarization. Thinning reduced the average to 18.82 points.

Termination Points (Original/Normalized): Average of 30.72 points before enhancement, increased to an average of 31.18 after normalization, and maintained at an average of 27.94 after binarization. Thinning reduced the average to 30.06 points.

V. CONCLUSION

As shown in the results of this experiment, enhancement techniques applied to fingerprint

images did improve the effectiveness. Normalization, binarization, Log-Gabor filtering, and Fourier Transform were found to improve the clarity of the images so much that the minutiae points became more observable and less difficult to find. The contribution of Gaussian filtering and contrast enhancement was made to clear the image from noise and to make more outstanding the important ridge structures themselves.

Combining all these enhancement techniques, the system has benefited from improved clarity of fingerprint images, which is essential for effective minutiae detection and verification of fingerprints. Further optimizations for reducing the processing times, especially for computationally expensive techniques such as Fourier Transform, while maintaining improvements in minutiae detection, can be areas in further work..

REFERENCES

[1] Hong, L., Wan, Y., & Jain, A. K. (1998). Fingerprint image enhancement: Algorithm and performance evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8), 777- 789.

[2] Shah, A., & Kothari, R. (2021). A Study of Fingerprint Image Enhancement Techniques. International Journal of Engineering Research & Technology (IJERT), 10(6), 1203-1211.

[3] Mehtre, B. M., & Chatterjee, B. (2011). Segmentation of Fingerprint Images Using the Fourier Transform. International Journal of Computer Applications, 975-8887.

[4] Tico, M., & Kuosmanen, P. (2003). An algorithm for fingerprint image postprocessing. Pattern Recognition, 36(6), 1259-1270.

[5] Belcher, C., & Du, Y. (2008). Region-based image enhancement: a novel fingerprint enhancement method using Fourier transform. IEEE Transactions on Image Processing, 22(6), 2351-2360.

[6] Bharati, M. H., & Liu, J. J. (2014). Fingerprint image enhancement by regional segmentation-based Fourier transform. International Journal of Biometrics, 6(2), 99-113.

[7] Ross, A., & Jain, A. K. (2013). Biometric Sensor Interoperability: A Case Study in Fingerprints. Proceedings of the IEEE, 2003-2021.

[8] Zhu, E., & Hu, C. (2019). A new method for fingerprint image enhancement using adaptive filtering in the Fourier domain. IEEE Access, 7, 12432-12441.

[9] Gonzalez, R. C., & Woods, R. E. (2002). Digital Image Processing. Prentice Hall.

[10] O'39;Gorman, L., & Nickerson, B. G. (1988). An approach to fingerprint filter design. Pattern Recognition, 22(1), 29-38.

[11] Nalini, H., & Yogish, H. (2017). Fingerprint Image Enhancement using Fourier Transform and Gabor Filter. International Journal of Engineering Trends and Technology (IJETT), 47(2), 102-106.

[12] Liu, Y., & Jiang, X. (2017). Fingerprint image postprocessing: A combined method of non-subsampled contourlet transform and Fourier transform. IET Biometrics, 6(4), 285-292.

[13] Wang, X., & Hu, J. (2019). A fast fingerprint enhancement algorithm based on mean curvature. IEEE Access, 7, 74251-74262.

[14] Chen, X., & Moon, Y. S. (2019). A novel algorithm for fingerprint image enhancement using spatial and frequency domain filtering. Applied Sciences, 9(10), 2074.

[15] Yang, J., & Liu, L. (2017). Local frequency descriptor based on discrete Fourier transform for robust multimodal biometric recognition. IEEE Access, 6, 14235-14246.

[16] Cao, K., & Jain, A. K. (2018). Automated Latent Fingerprint Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(4), 1023-1038.

[17] Ionescu, R. T., & Popescu, M. (2017). Local learning-based histogram representation for image enhancement. IEEE Transactions on Image Processing, 27(4), 1626-1637.

[18] NeerajBharagava, AnchalKumawat, RituBharagava (2015). "Fingerprint Matching of Normalized Image based on Euclidean Distance". International Journal of computer Application. Volume 120-No 24.

★★★