Adaptive Latent Fingerprint Segmentation using Feature Selection and Random Decision Forest Classification

Anush Sankaran, Aayush Jain, Tarun Vashisth, Mayank Vatsa, Richa Singh

Abstract

Latent fingerprints are important evidences used by law enforcement agencies. However, current state-of-the-art for automatic latent fingerprint recognition is not as reliable as live-scan fingerprints and advancements are required in every step of the recognition pipeline. This research focuses on automatically segmenting latent fingerprints to distinguish between ridge and non-ridge patterns. There are three major contributions of this research: (i) a machine learning algorithm for combining five different categories of features for automatic latent fingerprint segmentation, (ii) a feature selection technique using modified RELIEF formulation for analyzing the influence of multiple category features on latent fingerprint segmentation, and (iii) a novel SIVV based metric to measure the effect of the segmentation algorithm without the requirement to perform the entire matching process. The image is tessellated into local patches and saliency based features along with image, gradient, ridge, and quality based features are extracted. Feature selection is performed to study the contribution of the various category features towards foreground ridge pattern representation. Using these selected features, a trained Random Decision Forest based algorithm classifies the local patches as background or foreground. The results on three publicly available databases demonstrate the efficacy of the proposed algorithm.

Keywords: Saliency; Random Decision Forest; Feature Selection; Latent Fingerprint Segmentation

1. Introduction

Latent fingerprints are (un)intentional deposition of ridge patterns on a surface that comes in contact with a fingerprint. In law enforcement applications, latent fingerprints are used as a crucial forensic evidence for crime scene analysis. Existing semi-automated procedures for latent fingerprint matching consist of: (1) preprocessing, (2) manual annotation of features such as minutiae and singular points, (3) search for top-k probable matches (typically $k = 50$) using an Automated Fingerprint Identification System (AFIS), and (4) manual verification of the candidate list by forensic experts. The preprocessing stage consists of (i) quality check of latent fingerprints, (ii) segmenting the foreground fingerprint regions from the noisy background, and (iii) enhancing the clarity of ridge patterns. In current approaches, forensic experts have to manually pre-process a large number of fingerprints which is time consuming and not scalable. Therefore, automating these steps can significantly increase the efficiency of AFIS. With the same intent, FBI's Next Generation Identification (NGI) [1] aims at building an automated "lights-out" latent fingerprint matching system.

This research paper focuses on automating the task of latent fingerprint segmentation. As shown in Figure 1, some of the factors involved in making latent fingerprint segmentation a difficult problem are that fingerprints may be of poor ridge clarity [3] or certain latent fingerprints may not have a clear boundary due to smudges and background noise. A recent survey by Sankaran et al. [4] discusses the challenges involved in automatic latent fingerprint segmentation.

- **Poor ridge clarity:** The varying background surfaces, smudges, and non-linear distortion introduced while lifting, as shown in Figure 1(a), reduces the ridge information in the print.

Email address: {anushs, aayush09055, tarun09051, mayank, rsingh}@iiitd.ac.in (Anush Sankaran, Aayush Jain, Tarun Vashisth, Mayank Vatsa, Richa Singh)

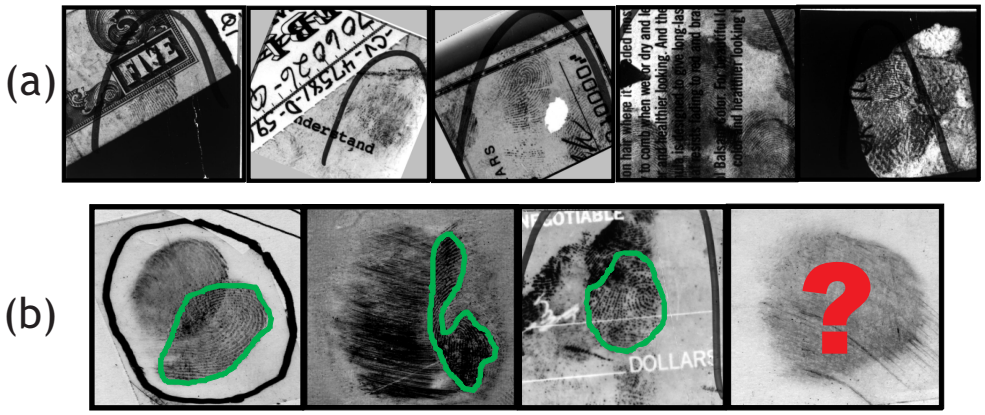

Figure 1: (a) Sample latent fingerprint images from the NIST SD-27 database [2] demonstrating the effect of background information on ridge information and (b) latent fingerprint samples illustrating the problem of segmentation.

- **Structured marks:** Structured markings such as arch, lines, and characters, as shown in Figure 2(a), have patterns that are similar to ridges, thereby adding to the challenges faced by an automatic latent fingerprint segmentation algorithm.
- **Optimal representation:** The output of fingerprint segmentation can be represented in multiple ways, as shown in Figure 3. Selecting the optimal representation with maximum information is an essential research challenge.
- **Performance metric:** The performance of the preprocessing stage is usually evaluated as an improvement in rank-k matching performance [5], [6]. However, as the overall latent print matching involves other complex stages like minutia extraction and matching, there is a desire to evaluate the success of preprocessing stage as-is.
- **Overlapped prints:** Two or more latent fingerprints may be overlapped, as shown in Figure 2(b). Estimating the orientation of latent fingerprints independently and segmenting individual fingerprints in such cases is also a difficult problem.

Some latent fingerprint segmentation approaches have been developed in literature which are summarized in Table 1. In 2008, Karimi and Kuo [7] proposed an automated latent fingerprint segmentation technique by measuring the variability in the ridge frequency and gradient in the local blocks. The performance of their algorithm was visually demonstrated using two images from the NIST SD-27 database [2]. In 2011, Short et al. [8] proposed a latent fingerprint segmentation technique by cross-correlating latent prints with an ideal template of ridge patterns. The correlation strength identifies ridge-like patterns thereby segmenting the foreground regions. Their algorithm yielded an Equal Error Rate (EER) of 33.8% on the NIST SD-27 database. Choi et al. [9] combined fingerprint orientation tensor and frequency tensor information to segment foreground from background. They showed a rank-1 performance accuracy of 16.28% on the NIST SD-27 and 35.19% on the WVU database. Their approach also yielded a Missed Detection Rate (MDR) of 14.78% and a False Detection Rate (FDR) of 47.99% in NIST SD-27 database. Zhang et al. [5] proposed an Adaptive Directional Total Variational (ADTV) model which is a variation of TV-L2 model. The ADTV model is suitable for decomposing textures with orientation patterns. The orientation pattern forms a defined structure in foreground ridge regions. Their approach yielded a Rank-1 identification accuracy of about 2% on NIST SD-27, with a MDR of 14.10% and FDR of 26.13%. Recently, Cao et al. [6] used a combination of coarse and fine structure ridge dictionary to learn a sparse representation of ridge-like patterns. Rank-1 identification accuracy of 61.24% was reported on NIST SD-27 and 70.16% on WVU database.

In this research, we propose a feature selection and learning based classification approach for segmenting fingerprint foreground from background. As shown in Figure 4, the interleaving ridge-valley patterns and the background are much clearer and distinct in live-scan fingerprints than in latent fingerprints. Therefore, any single feature or a category of features is unlikely to yield proper segmentation results across all kinds of latent fingerprints lifted from

Table 1: Review of existing latent fingerprint segmentation techniques.

Paper	Methodology	Database	Results	Other metrics
Karimi & Kuo [7], 2008	Variation in ridge frequency and gradient	2 images from NIST SD-27	-NA-	-NA-
Short et al. [8], 2011	Cross-correlation strength	NIST SD-27	Equal Error Rate of 33.8%	-NA-
Choi et al. [9], 2012	Orientation and frequency tensor	NIST SD-27 and WVU with extended gallery of 31997 images	Rank-1 accuracy of 16.28% on NIST SD-27 and 35.19% on WVU DB	Missed Detection Rate, False Detection Rate
Zhang et al [5], 2013	Adaptive directional total variational model	NIST SD-27 with extended gallery of 27258 images	Rank-1 accuracy of less than 3%	Missed Detection Rate, False Detection Rate, Feature extraction
Cao et al. [6], 2014	A coarse and fine structured ridge dictionary	NIST SD-27 and WVU database with extended gallery of 31997 images	Rank-1 accuracy of 61.24% on NIST SD-27 and 70.16% on WVU DB	-NA-

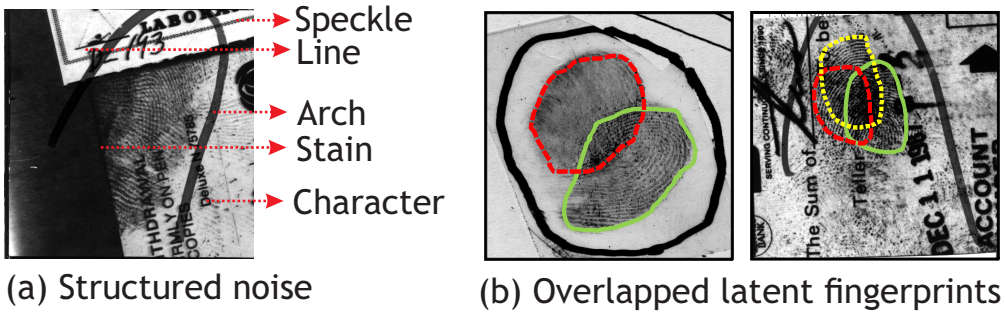

Figure 2: Sample fingerprint images from NIST SD-27 showcasing two specific challenges in latent fingerprint segmentation. (a) presence of structured noise in latent fingerprint background that often resembles ridge like patterns, and (b) overlapped fingerprints result in overlapped ridge information making it difficult to determine the ridge flow of either of the fingerprints.

different surfaces. Also, to accommodate the variations in the ridge patterns and to make generalized conclusions, the segmentation algorithm needs to select useful features and to learn the difference between background and foreground regions from these features. Inspired from these observations, the proposed approach extracts a composite set of features to represent latent fingerprint ridge patterns, performs feature selection, and classification for improved accuracy. The key research contributions of this research are:

- Latent fingerprint segmentation is modeled as a learning based two-class classification problem with *foreground* and *background* being the two classes. To the best of our knowledge, no classification based segmentation approach has been proposed for latent fingerprints. Though in live-scan fingerprints, there are classification based segmentation approaches [11, 12, 13, 14], the nature of latent fingerprints (with poor ridge features, varying background, overlapping foreground-background information) cause difficulty in applying existing algorithms on these images. The problem is further exacerbated with the availability of small sample size latent fingerprints databases where classes have high intra-class variations. Traditional classifiers such as Support Vector Machine (SVM) and Neural Network on such databases may not yield good classification results. Therefore, Random Decision Forest (RDF) classifier, which is an ensemble of multiple decision trees, is utilized to classify the extracted features into the two classes.

- We propose image saliency as a key feature for latent fingerprint ROI detection. To the best of our knowledge, saliency has not been used for fingerprint segmentation in literature. It is observed that salient regions of an image contain the foreground ridge information. However, for some instances, the background has very distinct, salient objects other than fingerprints. Therefore, along with saliency based features we combine image intensity

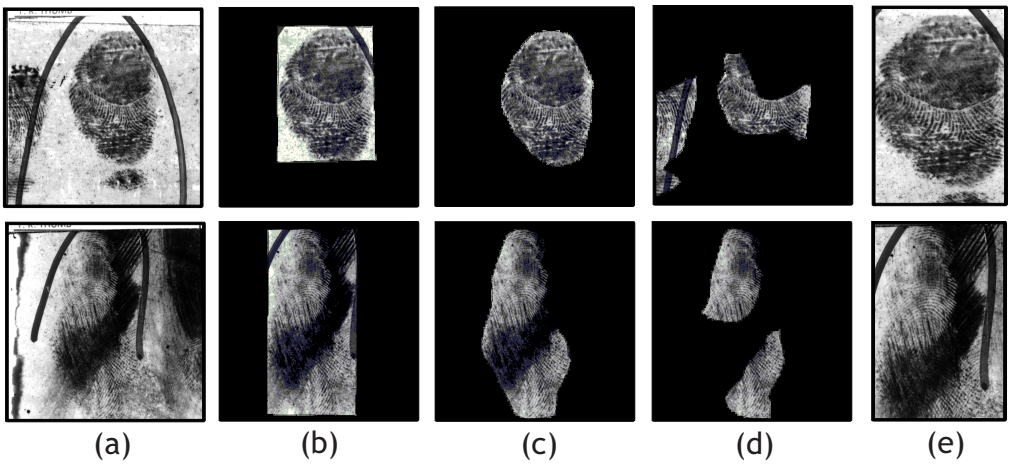

Figure 3: Sample latent fingerprints from the NIST SD-27 fingerprint database [2] showing segmentation results. (a) Original latent fingerprint images, (b) manually segmented output with a bounding box around the fingerprint region, (c) manually segmented output with exact boundary around the fingerprint region, (d) manually segmented output with only useful ridge information rejecting all the smudgy and noisy (non-informative) regions, and (e) segmented output from *NFSEG* module of NBIS [10].

based features and fingerprint specific features (gradient based, ridge based, and quality based) to obtain a more robust representation of the fingerprint ridge patterns. We grouped many of the existing features into five categories to perform a more system study on the foreground representation.

- A modified RELIEF formulation is proposed to perform feature selection and study optimal features for fingerprint segmentation which are finally used for classification.
- The performance of the proposed algorithm is evaluated using two different databases: (1) NIST SD-27 database [2] and (2) IIIT-D Combined Latent Fingerprint (CLF) database, which is a combination of IIIT-D latent fingerprint database [15] and IIIT-D SLF database [16]. The segmentation performance is evaluated in terms of multiple metrics: (1) Spectral Image Validation and Verification (SIVV) based metric for evaluating the effect of segmentation, (2) segmentation accuracy (SA) which captures the amount of useful information retained after segmentation, and (3) matching accuracy (MA) which captures the contribution of segmentation process in improving the latent fingerprint matching performance.

2. Proposed Segmentation Algorithm

Latent fingerprint segmentation is formulated as a binary classification problem where every local region is classified as either foreground or background. As illustrated in Figure 5, the proposed segmentation algorithm consists of local block tessellation, feature extraction, and feature selection followed by RDF based binary classification.

2.1. Feature Extraction

Determining whether a local block contains fingerprint patterns requires extracting patterns that are very specific to fingerprints (e.g. ridge patterns). A significant amount of research has undergone in describing ridge patterns or segmenting inked and live-scan fingerprints [11, 12, 13, 14]. The features used for distinguishing fingerprint foreground and background can be classified into five categories namely:

1. Saliency-based features: General saliency based features can be used to define the most salient regions in a latent fingerprint image.
2. Image intensity-based features: Features such as intensity mean, variance, and ridge cluster degree can be grouped into image intensity based features.

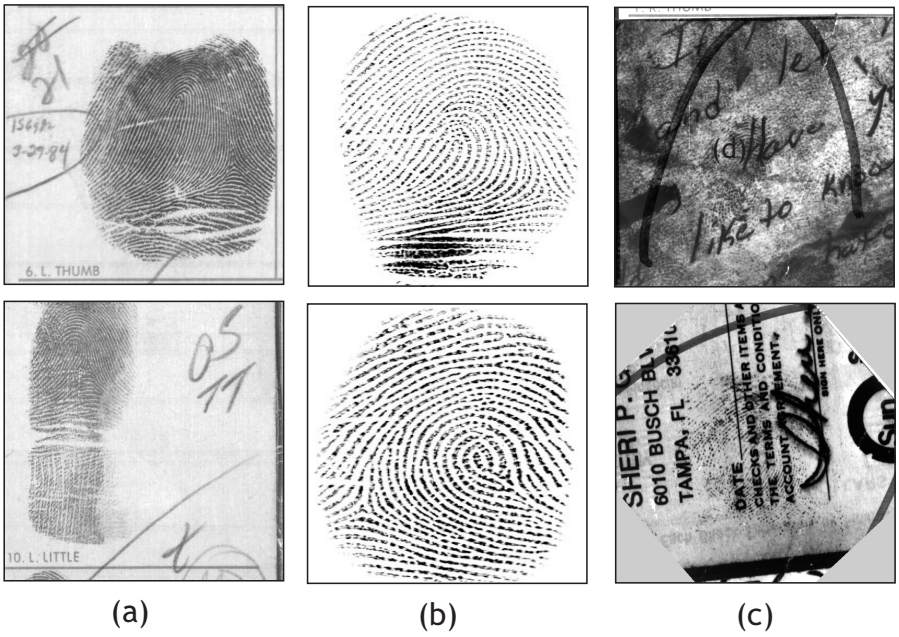

Figure 4: Sample latent fingerprints illustrating the distinct nature of foreground ridge patterns and background. (a) inked fingerprints, (b) live-scan fingerprints, and (c) latent fingerprints.

3. Gradient-based features: Features such as ridge orientation, variance along and normal to the ridge orientation flow, and symmetric orientation response can be categorized as gradient based features.

4. Ridge-based features: Features such as inter-ridge distance, ridge frequency, and ridge angular bandwidth belong to ridge based features.

5. Quality-based features: Features such as ridge energy and ridge continuity map measure the quality of local ridge blocks.

Let I be the input latent fingerprint image and $I(i, j)$ denote the intensity at pixel coordinates (i, j). The image is tessellated into local blocks of size $w \times w$ and five categories of features (mentioned above) are extracted from every local block.

2.1.1. Saliency-based features

In an image, a salient region is defined as the region which is noticed first by a human eye [17]. In general, salient regions are the most informative regions in an image and in the case of fingerprint, it is generally the foreground that we notice first. Therefore, saliency-based features are applied to segment foreground from background. Here, two such features are used, orientation and intensity of the intermediate neighborhood, that help in generating the saliency map of a fingerprint image. As studied by Harol et al. [18], a saliency map gives higher values in the most salient regions i.e. the fingerprint region along with some prominent background regions. Thus the intermediate features, intensity and orientation ($f_1 - f_2$), should give a similar higher response in the foreground as shown in Figure 6. The saliency features are computed as follows:

(i) **Salient Intensity (f_1)**: This feature is related to the saliency of a pixel, which is computed as the dissimilarity of the given pixel with respect to its $w \times w$ neighborhood (in terms of image intensity). The dissimilarity measure

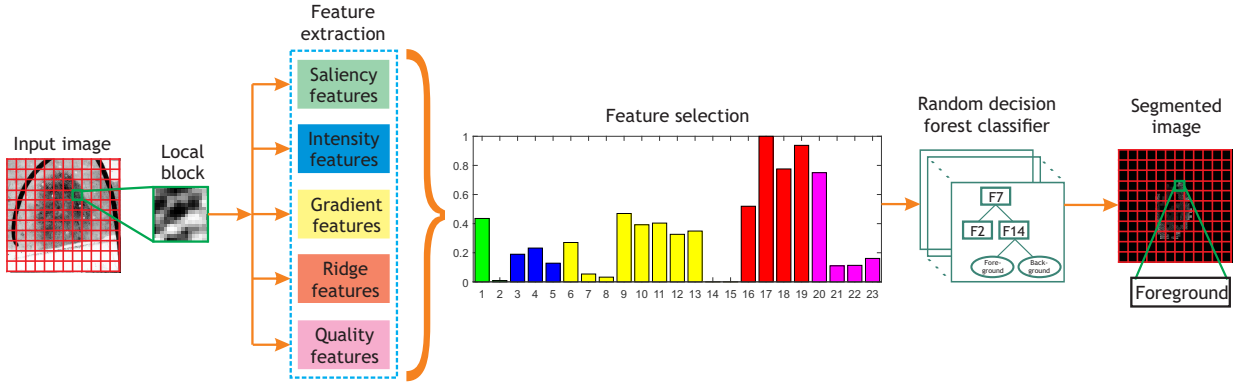

Figure 5: Illustrating the steps involved in the proposed RDF based latent fingerprint segmentation algorithm. A composite set of features is extracted from every local block and a random decision forest based binary classifier is used to classify the foreground regions from background.

is weighted by a Gaussian function,

$$f_1 = \sum_{i=-\frac{w}{2}}^{\frac{w}{2}} \sum_{j=-\frac{w}{2}}^{\frac{w}{2}} \left| \log \frac{I(x,y)}{I(x+i,y+j)} \right| \cdot \exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \tag{1}$$

where, (x, y) is the center pixel of the local block and σ is a free variable which is assigned the value 0.5. Higher value of dissimilarity in the local block represents that the region is more salient.

(ii) **Salient Orientation** (f_2): This feature is computed by the summation of the Gabor filter responses along two orientations: 0 degrees and 90 degrees. The orientation feature is calculated as follows:

$$f_2 = abs(F \bullet G(0)) + abs(F \bullet G(90)) \tag{2}$$

where, F is the Fourier transform output of the local block. $G(0)$ and $G(90)$ are the Gabor filters along zero degrees and 90 degrees, respectively and $< \bullet >$ denotes 2D convolution.

Though saliency extraction algorithm [18] is designed for natural images, we observe that the approach provides useful saliency maps in fingerprints as well, that can be used for segmentation. It is to be noted that saliency of all the pixels are computed and then block-wise features are extracted.

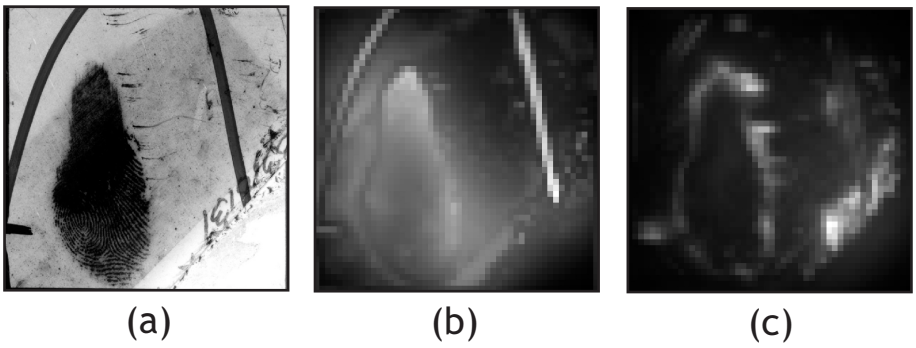

Figure 6: (a) A sample image from the NIST SD-27 database with corresponding saliency-based (b) intensity feature (f_1), and (c) orientation feature (f_2).

2.1.2. Image intensity-based features

In a latent fingerprint image, the variation in intensity values is usually definite along the ridges and valleys when compared to noisy background regions. Further, as shown in Figure 7, the properties of image intensity in a local region of the foreground is different from the image intensity of the background. These properties are extracted using three different intensity-based features.

(i) **Difference of means (f_3)** computes the difference between the local intensity mean and the global intensity mean. As a result of varying intensities in the background and foreground, the global intensity mean would be closer to average grayscale value. For a local foreground fingerprint region, due to the interleaved ridge-valley structures, the mean intensity value would be closer to average grayscale value of the image than compared to a background region. Therefore, ideally the difference of means should be lower in the foreground as compared to background.

$$f_3 = \left(\frac{1}{w^2} \sum_{i=1}^{w} \sum_{j=1}^{w} I(i,j) \right) - I_{mean} \tag{3}$$

where I_{mean} is the mean intensity of the complete image.

(ii) **Variance (f_4)** calculates the intensity variance in a $w \times w$ image block. Since the variance in an interleaved ridge-valley structure would be higher, high variance is expected in a fingerprint region as compared to background.

$$f_4 = \frac{1}{w^2} \sum_{i=1}^{w} \sum_{j=1}^{w} \left(I(i,j) - \frac{1}{w^2} \sum_{i=1}^{w} \sum_{j=1}^{w} I(i,j) \right)^2 \tag{4}$$

(iii) **Ridge cluster value (f_5)** indicates the clustering between the ridge pixels. This feature combines the properties of both mean and variance to capture the ridge-valley structure in a fingerprint foreground region. It can be calculated as follows [19]:

$$f_5 = \sum_{i=1}^{w} \sum_{j=1}^{w} v_1(i,j) \times v_2(i,j) \tag{5}$$

where

$$v_1(i,j) = \begin{cases} 1 & \text{if } I(i,j) < I_{mean} \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

$$v_2(i,j) = \begin{cases} 1 & \text{if } s(i,j) < \left(\frac{n^2}{2} + 1 \right) \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

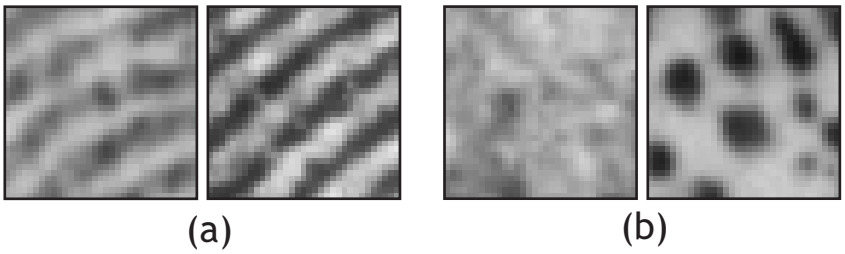

Figure 7: Sample local blocks from the NIST SD-27 database with (a) foreground ridge blocks and (b) noisy background blocks. Varying image intensity patterns can be observed between the foreground and the background blocks.

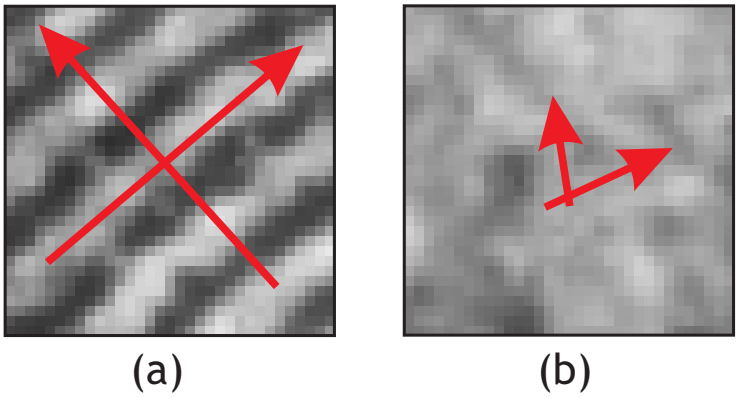

Figure 8: Sample local square blocks from the NIST SD-27 database with (a) dominant orthogonal orientation observed in a foreground ridge block, (b) no dominant direction found in a noisy background block.

$$s(i,j) = \sum_{x=i-\frac{n}{2}}^{x=i+\frac{n}{2}} \sum_{y=j-\frac{n}{2}}^{y=j+\frac{n}{2}} v_1(x,y) \tag{8}$$

Here, I_{mean} is the global mean intensity and the number of pixels s, in the neighborhood $n \times n$ that have intensity lower than I_{mean} (typically valley regions) is measured. s tends to be larger in uniform background regions than in ridge-valley like regions. Feature f_5 measures the number of pixels in a local block $w \times w$ whose s value is below a specific threshold. Thus f_5 maybe a robust measure even in regions with broken or noisy ridges.

2.1.3. Gradient-based features

The gradient of an image is used to capture the directional change in pixel intensities along a direction. This change in directional flow will be more regular in a fingerprint region as compared to background where the noise gives a non-directional change in the flow [20]. Gradient is also being used to estimate the orientation of ridges in local regions as shown in Figure 8. For a latent fingerprint image I, let $[I_x, I_y]$ be the gradient along x and y directions, respectively. The orientation at location (i,j) is calculated as:

$$O(i,j) = \begin{cases} \pi/4 & c_1 = 0, c_2 < 0 \\ 3\pi/4 & c_1 = 0, c_2 \geq 0 \\ \theta^I(i,j) + \pi/2 & c_1 > 0 \\ \theta^I(i,j) & c_1 < 0, c_2 \leq 0 \\ \theta^I(i,j) + \pi & c_1 < 0, c_2 > 0 \end{cases} \tag{9}$$

$$\theta^I(i,j) = \frac{1}{2} tan^{-1}\left(\frac{c_2}{c_1}\right) \tag{10}$$

where, c_1 and c_2 are defined as follows:

$$c_1 = \sum_{i=1}^{w}\sum_{j=1}^{w} \left(I_x^2(i,j) - I_y^2(i,j)\right) \tag{11}$$

$$c_2 = \sum_{i=1}^{w}\sum_{j=1}^{w} 2 \cdot I_x(i,j) \cdot I_y(i,j) \tag{12}$$

The gradient properties are formulated using six different features that are explained below.

(i) **Ridge orientation (f_6)** is computed by smoothing the orientation over the block using a Gaussian smoothing kernel [21].

$$f_6 = \frac{1}{w^2} \sum_{i=1}^{w} \sum_{j=1}^{w} O'(i,j) \tag{13}$$

$$O'(i,j) = \frac{1}{2} tan^{-1} \left(\frac{sin(2O(i,j)) * G(i,j)}{cos(2O(i,j)) * G(i,j)} \right) \tag{14}$$

where, G is the Gaussian smoothing kernel of size 3×3 and $O(i,j)$ is defined in Equation 9.

(ii) **Sum of squared gradient (f_7)** represents the sum of squares of the gradient values of a local block. The interleaving ridge-valley pattern provides a change in flow that is higher as compared to the noisy background.

$$f_7 = \sqrt{c_1^2 + c_2^2} \tag{15}$$

(iii) **Sum of norm of squared gradient vector (f_8)** is computed as:

$$f_8 = \sum_{i=1}^{w} \sum_{j=1}^{w} \sqrt{\begin{matrix} \left(I_x^2(i,j) - I_y^2(i,j)\right)^2 + \\ \left(2 \cdot I_x(i,j) \cdot I_y(i,j)\right)^2 \end{matrix}} \tag{16}$$

(iv) **Variance of projected axis parallel to orientation (f_9)** is calculated by computing the ridge variation in the direction parallel to the estimated local block orientation. A projection window of size $B \times H$, which is smaller than the block size and whose center overlaps with center of the local block (k,l) is used to compute f_9:

$$f_9 = \sum_{l=-B/2}^{B/2} \left(Pv[l] - \sum_{k=-B/2}^{B/2} Pv[k]/B \right)^2 \tag{17}$$

where,

$$\begin{aligned} Pv[k] = & \frac{1}{H} \sum_{h=-H/2}^{H/2} I(i - h \cdot sin(O(i,j)) + k \cdot cos(O(i,j)), \\ & j + h \cdot cos(O(i,j)) + k \cdot sin(O(i,j))) \end{aligned} \tag{18}$$

(v) **Variance of projected axis orthogonal to orientation (f_{10})** is calculated by computing ridge variation in the direction normal to the estimated local block orientation. Similar to f_9, a projection window of size $B \times H$ perpendicular to the estimated orientation of ridges is considered and the features are computed as follows:

$$f_{10} = \sum_{l=-B/2}^{B/2} \left(Ph[l] - \sum_{k=-B/2}^{B/2} Ph[k]/B \right)^2 \tag{19}$$

where,

$$\begin{aligned} Ph[k] = & \frac{1}{H} \sum_{h=-H/2}^{H/2} I(i + h \cdot cos(O(i,j)) + k \cdot sin(O(i,j)), \\ & j + h \cdot sin(O(i,j)) - k \cdot cos(O(i,j))) \end{aligned} \tag{20}$$

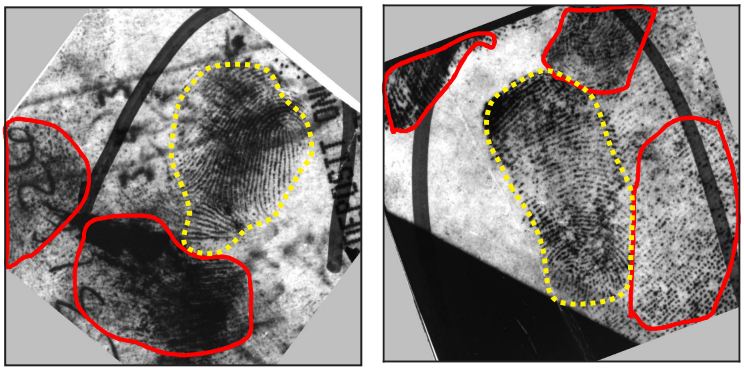

Figure 9: Sample images from the NIST SD-27 database showing noise in latent fingerprint images. The yellow dotted lines are the actual fingerprint regions while the red full lines are ridge like noisy regions in the background.

As suggested by Zhu et al. [14], a projection window of size 12×9 has been used for calculating both f_9 and f_{10}. In foreground ridge-like regions, f_9 exhibits very low variance whereas f_{10} exhibits high variance. In background regions, f_9 and f_{10} remain almost constant without much variation.

(vi) **Mean of symmetry and texture patterns ($f_{11} - f_{15}$):** Let the complex representation of an image be denoted as $z = I_x + iI_y$. As shown by Choi et al. [9], the n^{th} order symmetric decomposition of the orientation response of an image block can be computed using:

$$[f_{11}, f_{12}, f_{13}, f_{14}, f_{15}] = \{s_0, s_1, s_{-1}, s_2, s_{-2}\} \tag{21}$$

$$s_n = \frac{< z, h_n >}{< abs(z), abs(h_n) >} \tag{22}$$

$$h_n = \begin{cases} (x+iy)^n \bullet G & \text{if } n \geq 0 \\ (x+iy)^{|n|} \bullet G & \text{otherwise} \end{cases} \tag{23}$$

where, G is the Gaussian filter with $\sigma = 8$ and $< \bullet >$ denotes 2D convolution. The orientation response of an image block is decomposed into five symmetric orders providing features $f_{11} - f_{15}$ for $n = 0, \pm 1, \pm 2$ respectively. The peak response for s_0 is obtained in foreground ridge-like regions whereas s_1, s_{-1}, s_2, and s_{-2} give peak response in the background regions [22].

2.1.4. Ridge-based features

As shown in Figure 9, a latent fingerprint may contain many ridge like noisy patterns belonging to other fingers in the background. To differentiate the actual fingerprint from such noisy patterns, the properties of ridges are extracted to effectively test the presence of ridge patterns [23]. The four different features that have been utilized in this research to encode ridge information are:

(i) **Average inter-ridge distance (f_{16}):** Ridge peaks in the local block are computed using the gradient approach [12]. The mean of absolute difference between any two consecutive peaks is denoted as f_{16} and is computed as follows:

$$f_{16} = \frac{\sum_{k=1}^{n} a_k}{n-1} \tag{24}$$

where, n is the number of peaks in the ridges and a_k is the distance between two consecutive peak values. As

the number of ridges is higher in a fingerprint region, the inter-ridge distance here would be less as compared to background.

(ii) **Variance of peak heights in ridges (f_{17}):** It estimates the variations in ridge pressure that can be observed in a local block of a latent fingerprint. It can be computed as follows:

$$f_{17} = \frac{\sum_{k=1}^{n}(PR_k - PR_{mean})}{n-1} \tag{25}$$

where, n is the number of peaks in the ridges, PR_k is the value of the peak ridge height for the k^{th} ridge, and PR_{mean} is the mean of the peak ridge heights across all the blocks. A higher response is expected in a fingerprint region as compared to a non-fingerprint region.

(iii) **Ridge frequency (f_{18})** is calculated by applying Fourier transform to every local block, commonly known as Short Time Fourier Transform (STFT) [21]. The magnitude spectrum of frequency response is multiplied with a set of directional filters with varying frequencies. The frequency of the filter at which the maximum response is obtained, is considered to be the ridge frequency of the local block.

$$f_{18} = argmax_l \left(\sum_{u=1}^{w} \sum_{v=1}^{w} |F(u,v)| * W_l(u,v) \right) \tag{26}$$

where, $F(u,v)$ is the Fourier transform output of the local image block and $W_l(u,v)$ is the l^{th} directional filter. Since a ridge-valley structure can be interpreted as a sinusoidal structure, the frequency response is higher in a structured fingerprint region as compared to a noisy and unstructured background region.

(iv) **Angular bandwidth (f_{19})**: Similar to f_{18}, STFT is applied and peak response is calculated for every block. The bandwidth of directional filter along the local estimated orientation that provides the peak response is the angular bandwidth of ridges for the local block.

2.1.5. Quality-based features

Assessing the quality information in a local fingerprint region is very useful for segmentation. The quality of ridge patterns acts as a measure of confidence of the features extracted in the local region. Foreground regions should have a higher quality information compared to the noisy background regions. The quality features are extracted as follows:

(i) **Ridge energy (f_{20})**: The STFT response of a local fingerprint block is subjected to a band-pass filter allowing only the specified ridge frequencies to pass [24]. The ridge energy is computed as follows:

$$f_{20} = \frac{1}{w^2} \left(\sum_{u=1}^{w} \sum_{v=1}^{w} (|F(u,v)| \bullet W_{l'}(u,v))^2 \right) \tag{27}$$

where, $W_{l'}(u,v)$ is the l' directional filter giving the highest response. The ridge energy provides the "ridgeness" of the local region and is expected to be higher in a fingerprint region.

(ii) **Ridge energy after clustering (f_{21})**: The Fourier response of a local block is initially clustered into two regions using *k*-means clustering and smoothed using a Gaussian filter [25]. Then, similar to f_{20}, energy in a local block is calculated.

(iii) **Ridge continuity map (f_{22}):** Every local block is modeled with two 2D sine waves, $\mathcal{S}_1$ and $\mathcal{S}_2$, corresponding to the top two local amplitude maxima of ridge intensity [26]. An indicator function is created to check if the waves in consecutive blocks (in a 8-neighborhood condition) are continuous.

$$I_c(\mathcal{S}_1, \mathcal{S}_2) = \begin{cases} 1 & \text{if } \mathcal{S}_1,\mathcal{S}_2 \text{ are continuous} \\ 0 & otherwise \end{cases} \tag{28}$$

The ridge continuity map is then calculated as

$$f_{22} = \sum_{\mathcal{S}'_i \in N} max\{I_c(\mathcal{S}_1, \mathcal{S}'_1), I_c(\mathcal{S}_1, \mathcal{S}'_2)\} \tag{29}$$

where $1 \leq N \leq 8$ and $\mathcal{S}'_i$ are the blocks belonging to the neighborhood N. This ridge continuity measurement gives a higher response in a good quality fingerprint region.

(iv) **Ridge clarity map** (f_{23}): Ridge clarity map can be calculated by multiplying the peak magnitude value of every local block, a_1, with the corresponding ridge continuity map value. The response of ridge clarity map is high in a good quality fingerprint region but is robust against background patterns that look similar to ridge-valley patterns.

$$f_{23} = a_1.f_{22} \tag{30}$$

Thus, a composite set of 23 features $\{f_1, f_2, \ldots, f_{23}\}$ is utilized for differentiating the foreground ridge patterns from a (noisy) background. A summary of all the category-wise features is provided in Table 2. It is our hypothesis that image saliency potentially detects the latent print region in the image along with few other salient regions. Thus, when saliency features are combined with fingerprint specific features, the false positive regions could be minimized resulting in only the required ROI.

Table 2: Summary of features used to represent the foreground ridge features.

Saliency	**Image intensity**	**Gradient**	**Ridge**	**Quality**
f_1 Intensity f_2 Orientation	f_3 Difference of mean f_4 Variance f_5 Ridge cluster value	f_6 Ridge orientation f_7 Sum of squared gradient f_8 Sum of norm of squared gradient f_9 Variance of projected axis parallel to orientation f_{10} Variance of projected axis orthogonal to orientation f_{11} Mean of symmetry and texture pat- f_{15} terns	f_{16} Average inter-ridge distance f_{17} Variance of peak heights in ridges f_{18} Ridge frequency f_{19} Angular bandwidth	f_{20} Ridge energy f_{21} Ridge energy after clustering f_{22} Ridge continuity map f_{23} Ridge clarity map

2.2. Feature Selection

The proposed algorithm utilizes an aggregation of 23 features. However, not all of them are equally distinctive and can differentiate between foreground and background efficiently. Therefore, in the proposed algorithm, we perform feature selection to select highly discriminative features so that the classification algorithm provides improved (and meaningful) output. The effectiveness of the extracted features is evaluated individually for segmentation. Choosing a subset of relevant features for better performing the task at hand is a challenging research problem [27], [28]. In a binary classification setting, RELIEF [29] is a noise-tolerant, linear time feature selection algorithm that gives good results in the presence of higher training instances. The main advantage of RELIEF feature selection is its simplicity and it does not depend on any heuristics or assumptions. Let W be the weight vector calculating the relevance of each feature i. The standard RELIEF feature selection method is given as follows:

$$W_i = W_i - (X_i - NH_i)^2 + (X_i - NM_i)^2 \tag{31}$$

where, X_i refers to the i^{th} training instance, NH_i is the "near-hit" instance of i denoting the nearest neighbor of X_i that belong to the same class of X_i, while NM_i is the "near-miss" instance denoting the nearest neighbor of X_i belonging to its opposite class. Here, the nearest neighbor is calculated using Euclidean distance measure. It can be understood that the relevance of the weight value reduces if the near-hit of a particular point is at farther distance compared to its near-miss neighbour.

As studied by Robnik-Šikonja and Kononenko [30], RELIEF formulation can be optimized and modified by doing the following:

1. Use ℓ_1-norm to find the neighbours of X_i instead of using Euclidean distance
2. Calculate the absolute difference between the points, instead of squared difference
3. Choose k-nearest neighbours of an instance X_i instead of the single nearest neighbour.

The modified formulation of RELIEF feature selection used in this experiment is as follows:

$$W_i = W_i - \sum_{p=1}^{k} |X_i - NH_i^{(p)}| + \sum_{q=1}^{k} |X_i - NM_i^{(q)}| \tag{32}$$

where, $|.|$ represents the absolute difference between the features, NH is the near-hit vector denoting the *k*-nearest neighbours of X that belong to the same class of X, while NM is the near-miss vector denoting the *k*-nearest neighbours of X belonging to its opposite class. In our experiments, $k = 20$ is empirically observed to be optimal. A threshold is empirically applied on the weight vector and all the features contributing more than this threshold are considered in the optimal set of features for segmenting latent fingerprints.

2.3. Classification using RDF

A non-linear classification algorithm should potentially produce a sophisticated classification boundary between $\{$*background, foreground*$\}$ using the extracted feature. In this approach, every local block in a latent fingerprint is classified into foreground and background using Random Decision Forest [31]. RDF is a non-linear ensemble classifier consisting of multiple decision trees. It has been shown in literature that RDF yields good classification results for high dimensional data [32], [33]. NIST uses RDF as the classifier in their well-received latent print quality assessment algorithm NFIQ-2 [34]. The repetitive random sub-sampling strategy employed by RDF helps in providing robust and quicker results for overlapping features. Let N be the total number of data points, M be the number of predictor variables (features), and C be the total number of classes in a given data. A forest containing T trees is trained as follows:

1. For a ratio r ($0.5 < r \leq 1$), several bootstrap aggregates, each of size $r.N$, are created with replacement from the data.
2. Every decision tree, t, in the forest is trained with a single bootstrap of the data, thus creating an ensemble of classifiers.
3. At every node in the decision tree, a random feature sample, m (typically $m = \sqrt{M}$) is used to take the split decision based on an objective function.
4. Class labels c ($c \in \{1, \ldots, C\}$) are assigned to leaf nodes depending on the label associated with the corresponding training sample. Collision resolution techniques can be used if a particular leaf node receives multiple class labels through multiple paths.

An input test sample is classified using the trained classifier as follows:

1. The candidate set of features extracted from local blocks of a latent fingerprint are provided as input to the RDF.
2. Every individual decision tree, t, predicts a class label, ot_i, through repeated sub-sampling of features at every node.
3. The final predicted class label, p, is obtained from the ensemble of classifiers using a majority voting technique.

$$p = argmax_{c_j \in C} \left[\frac{\sum_{i=1}^{T} (ot_i == c_j)}{T} \right] \text{ for j=\{1,2,...,C\}.} \tag{33}$$

In the RDF implementation, for classifying $C = 2$ classes, $T = 1000$ independent decision trees are created with a bootstrap ratio of $r = 0.66$. At every node in a decision tree, $m = 5$ features are randomly sampled from $M = 23$ features.

2.4. Representing Segmented Latent Prints

The output of a segmentation algorithm can take multiple forms as shown in Figure 3. It is important to define a standard representation scheme for segmentation of latent fingerprints. In this research, we represent segmentation as a single n-degree polygonal boundary of the region of interest. The size of the output image is kept same as the input, with all the background information blacked out. Even if the background contains other partial fingerprints (ridge patterns), they are still treated as background and thus removed. To achieve such a standard representation, as shown in Figure 10, the following post processing steps are performed for the masks obtained from the classifier output:

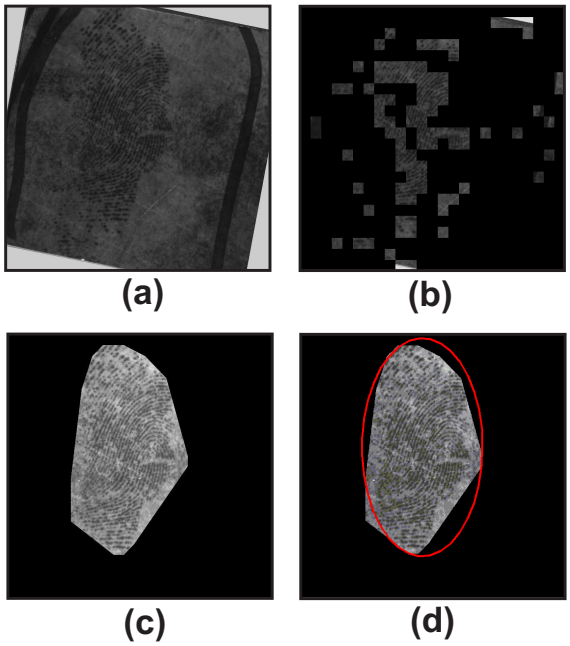

Figure 10: An example showing the post-processing performed on the classification output to arrive at the final segmentation output. (a) original input latent fingerprint, (b) classifier output of segmentation, (c) final segmentation boundary obtained after post-processing, and (d) an elliptical window fitted over the segmented region.

- The predicted classifier output contains blocks predicted as foreground or background, as shown in Figure 10 (b). Two iterations of morphological erosion are applied using a square structuring element of size same as the local block size. This helps in removing the false positives, that is, the background blocks that are classified as foreground.
- The largest connected component region in the image is then found using the standard run-length encoding technique [35]. Only the largest connected region is retained while the remaining blocks are regarded as background.
- A convex hull is fitted over the largest connected region, which gives the polygonal mask for the segmented region. The entire region within this boundary is filled as foreground to obtain the processed segmented output of the latent fingerprint, as shown in Figure 10 (c).

The processed segmented latent fingerprint acts as a better input for an AFIS system. It is to be noted that in special applications where the background ridge-like regions are required as well, the post processing stage can be skipped and the output of the classifier can be taken as such.

3. Evaluation Metrics for Latent Fingerprint Segmentation

The efficacy of a fingerprint segmentation algorithm is generally evaluated using matching accuracy. However, matching performance does not completely capture the performance of the segmentation algorithm as it includes the performance of other stages such as feature extraction as well. In cases where the ground truth segmentation is available, we propose to use segmentation accuracy (SA) to determine the effectiveness of a segmentation algorithm. Segmentation accuracy is represented in terms of the foreground segmentation accuracy (FSA) and background segmentation accuracy (BSA).

As ground truth of segmentation is not always available in practical situations, there is a need for a metric to evaluate the segmentation algorithm without ground truth. A recent NIST report [3] and the work by Guan et al. [36] have proposed a new metric for evaluating the effect of preprocessing on latent fingerprints based on Spectral Image Validation and Verification (SIVV) [37]. SIVV based True-Positive Rate (SIVV-TPR) metric finds the number of correct peaks detected in the 1-D normalized polar transform of the power spectrum of the latent print. A peak does not randomly occur in the frequency spectrum. As proposed in [36], a peak constraint metric is included to search for the peaks in a specific bandwidth in the frequency spectrum, to minimize the detection of false positive peaks. Thus, this metric could potentially evaluate the performance of the preprocessing stage, without performing the entire matching procedure. The major limitations of this metric are:

- **Shape of ROI**: A rectangular ROI around the latent print region has to be manually chosen by an expert. A tighter boundary estimate, extracted automatically, will provide a better segmentation of latent fingerprints.
- **Shape of Blackman window**: A circular Blackman window filter is applied prior to spectral analysis. Typically, a latent fingerprint is elliptical in shape and thus a circular filter will result in loss of information.

To address these limitations, we have proposed the following modifications to the SIVV metric: (i) a polygonal ROI is used instead of a rectangular ROI to represent the segmented latent print, and (ii) an elliptical Blackman window is used for filtering instead of a circular filter. Thus, the metrics used to measure the performance of latent fingerprint segmentation are:

1. SIVV based True-Postive Rate (SIVV-TPR) metric [3] is defined as:

$$SIVV - TPR = \frac{\text{Number of correct peaks detected}}{\text{Total number of peaks detected}} \tag{34}$$

2. Segmentation accuracy is the ability of a classifier to correctly classify image blocks into foreground and background. It is defined as:

$$SA = \frac{CCB}{TB} \tag{35}$$

where, CCB is the number of correctly classified blocks and TB is the total number of blocks. FSA can be calculated as,

$$FSA = \frac{CCFB}{TFB} \tag{36}$$

where, $CCFB$ is the number of correctly classified foreground blocks and TFB is the total number of foreground blocks in the ground truth image. Similarly, background segmentation accuracies can be calculated as,

$$BSA = \frac{CCBB}{TBB} \tag{37}$$

where, $CCBB$ is the number of correctly classified background blocks and TBB is the total number of background blocks in the ground truth images.

3. Matching accuracy is the fraction of latent images that are correctly identified at a given rank k, when matched with the corresponding gallery of live-scan images.

4. Experimental Results

The performance of the proposed segmentation algorithm is evaluated on three publicly available latent fingerprint databases. The algorithm is also analyzed to determine the optimal set of features that would best discriminate the ridge regions from the remaining background. The databases, evaluation metrics, and experimental protocol are described below along with the results.

4.1. Datasets

The results are shown on an inked fingerprint database and three publicly available latent fingerprint databases:

- NIST SD-4 database [38] is an inked fingerprint database consisting of 2000 rolled fingerprints pairs having very high quality ridge information with very minimum background variation.
- NIST SD-27 database [2] consists of 258 latent fingerprint images grouped into three quality labels: Good, Bad, and Ugly. It has mated rolled fingerprints for every latent print and also contains manually annotated minutiae for latent fingerprints.
- IIIT-D Latent Fingerprint database [15] has 744 latent impressions from 11 subjects (all 10 fingers) with mated live-scan fingerprints.
- IIIT-D Simultaneous Latent Fingerprint (SLF) database [16] has 1080 latent impressions from 30 subjects (all 10 fingers) with mated exemplar prints.

Since both the IIIT-D databases have been collected under similar environments, they are combined to form the IIIT-D Combined Latent Fingerprint (CLF) database. After combining, it consists of 1824 latent images from 41 subjects with multiple impressions of each finger. In these two sets of latent fingerprint databases, NIST SD-27 has real forensic fingerprints with high variation in quality whereas IIIT-D CLF database has large number of fingerprints collected in simulated lab environments.

Table 3: Experimental protocol for the NIST SD-4, NIST SD-27, and IIIT-D CLF databases. Inked fingerprints from 2000 classes of the NIST SD-4 database are added to extend the gallery.

Database	Training Dataset	Testing Dataset	Gallery Dataset	Image Size	Block Size
NIST SD-4	1000	1000	-	832×768	32×32
NIST SD-27	129	129	$258 + 2000$	800×768	32×32
IIIT-D CLF	544	1280	$820 + 2000$	256×400	16×16

4.2. Experimental Protocol

The experimental protocol is shown in Table 3. Since the NIST SD-27 database contains only 258 samples, 50% training and 50% testing protocol is followed. For the IIIT-D CLF database, a more challenging protocol of using one-third images for training and the remaining for testing is adopted. NIST SD-4 also uses a 50-50% train-test protocol. Due to the variations in image resolution in the databases, inked prints from NIST SD-4 and latent prints from the NIST SD-27 are divided into blocks of size 32×32 while the images from the IIIT-D CLF database are divided into 16×16 blocks. The ground truth for segmentation is manually annotated for all the latent prints from both the databases. A $n-$point contour is marked tangential to the foreground ridge region thus obtaining a binary mask[1]. The binary mask is then tessellated into square blocks and ground truth label for each block is assigned. In 2012, Ulery et al. [39] have suggested that since latent examiners use their subjectivity and experience in latent fingerprint analysis, the results tend to vary among human experts and may not be always reproducible. However, it is our assumption that manually annotated segmentation results should not vary significantly across examiners and therefore the performance analysis of the proposed algorithm also should not vary much with the variations in ground truth. To remove any training bias, three times random cross validation is performed on both the databases. The segmentation experiments are performed under the following scenarios:

[1]The manually annotated segmented binary masks will be made publicly available for researchers through the following link: http://iab-rubric.org/resources.html

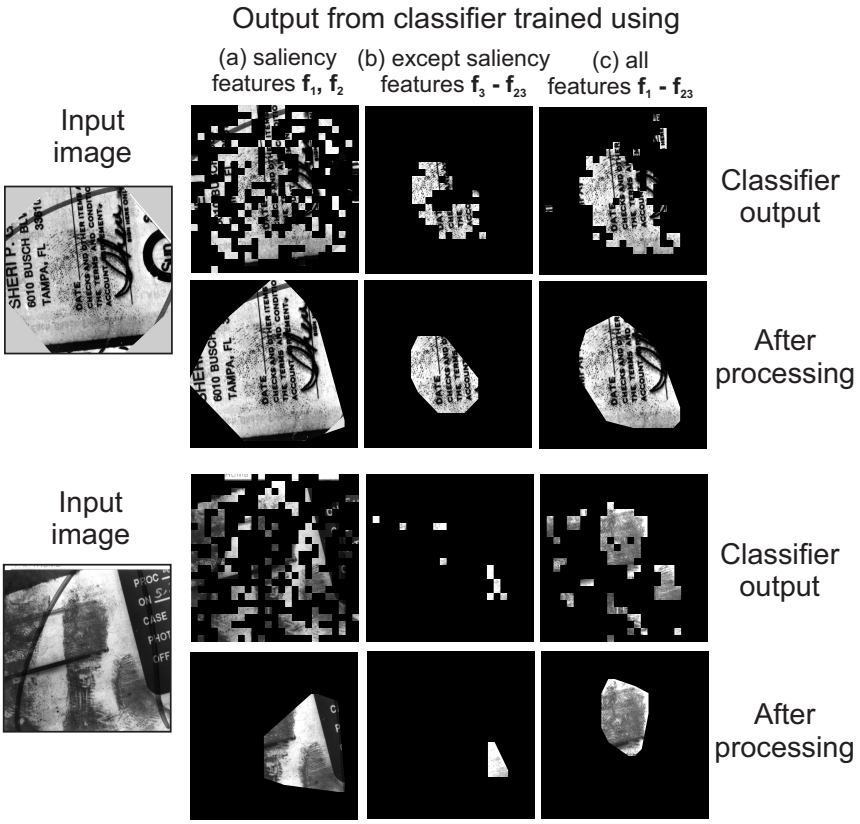

Figure 11: Illustrating the segmentation result of the proposed algorithm using two sample images from NIST SD-27 by using (a) only saliency features (f_1, f_2), (b) all other features except saliency ($f_3 - f_{23}$), and (c) all features (f_1, f_{23}).

1. Results of all the features ($f_1 - f_{23}$) are shown using the proposed RDF based classification algorithm.
2. Results of only the saliency features ($f_1 - f_2$) are shown using the proposed RDF based classification scheme.
3. Results of the selected optimal features (varying size for each dataset) are shown using the proposed RDF based classification scheme.
4. The effectiveness of RDF is demonstrated by comparing the performance of all the features ($f_1 - f_{23}$) with two other classifiers in literature - neural network and SVM. The neural network consists of a three hidden layer architecture with $\{20, 10, 5\}$ nodes each and a single output node with sigmoid activation function. SVM (libSVM implementation in MATLAB [40]) with a RBF kernel function ($c = 8$, $g = 2$) is found optimal.

Table 4: RELIEF algorithm based feature analysis on NIST SD-4, NIST SD-27, and IIIT-D CLF databases. The most and least contributing features for segmentation on each database are also obtained.

Database	**Best 3 features**	**Worst 3 features**	**Optimal features**
NIST SD-4	$\{f_1, f_{14}, f_{15}\}$	$\{f_{22}, f_2, f_{21}\}$	$\{f_1, f_{14}, f_{15}, f_{12}, f_{13}\}_5$
NIST SD-27	$\{f_1, f_6, f_{19}\}$	$\{f_4, f_2, f_{10}\}$	$\{f_1, f_6, f_{19}, f_5, f_{18}, f_{22}\}_6$
IIIT-D CLF	$\{f_4, f_5, f_1\}$	$\{f_3, f_{19}, f_6\}$	$\{f_4, f_5, f_1, f_9, f_{10}, f_8, f_{17}, f_{16}, f_7, f_{23}, f_{22}, f_{20}, f_{14}, f_{12}, f_{15}, f_{13}, f_{21}, f_{18}, f_2, f_{11}\}_{20}$

Table 5: Segmentation accuracy (SA), FSA and BSA (in %) of the proposed and existing segmentation algorithms on the NIST SD-4, NIST SD-27, and IIIT-D combined latent fingerprint databases.

Database	Metric	All Features + SVM	All Features + NN	All Features + RDF	Saliency Features + RDF	Optimal features + RDF
NIST SD-4	SA	91.84 ± 0.2	95.10 ± 0.7	96.11 ± 0.3	78.25 ± 0.2	91.90 ± 0.0
	FSA	92.54 ± 0.3	98.94 ± 0.6	95.76 ± 0.2	79.90 ± 0.7	92.49 ± 0.0
	BSA	91.37 ± 0.2	92.15 ± 0.6	96.35 ± 0.2	77.14 ± 0.5	91.50 ± 0.0
NIST SD-27	SA	66.24 ± 0.4	76.64 ± 0.2	73.76 ± 0.2	61.00 ± 0.1	66.35 ± 0.1
	FSA	78.45 ± 0.1	77.68 ± 0.1	83.41 ± 0.1	73.04 ± 0.1	85.11 ± 0.1
	BSA	63.18 ± 0.5	76.39 ± 0.2	71.34 ± 0.2	57.98 ± 0.1	61.63 ± 0.2
IIIT-D CLF	SA	89.33 ± 0.6	93.47 ± 0.2	93.57 ± 0.2	60.34 ± 0.6	93.47 ± 0.1
	FSA	91.59 ± 0.7	93.01 ± 0.1	93.23 ± 0.2	56.26 ± 0.5	93.01 ± 0.1
	BSA	87.41 ± 0.4	93.84 ± 0.1	93.84 ± 0.3	63.73 ± 0.7	93.84 ± 0.3

4.3. Importance of Saliency

Feature selection is performed separately on NIST SD-4, NIST SD-27, and IIIT-D CLF databases and the features contributing to better classification are tabulated in Table 4. The following important observations can be made based on the feature analysis:

- In all three databases, saliency (f_1) is one of the key features contributing towards segmentation. This aspect is relatively unexplored in the literature of latent fingerprints. If saliency features could be combined with fingerprint based features, a good representation of latent print foreground region can be obtained, as visually demonstrated in Figure 11.
- Apart from saliency, other features contributing towards ridge representation (best 3 features) are data dependent. This, as expected, explains the variation in ridge clarity in the databases used in this experiment.
- In inked fingerprints (for example images from the NIST SD-4 database), the texture pattern related features are more appropriate for segmentation while ridge continuity and energy based quality features contribute the least.
- In the most popular latent fingerprint database NIST SD-27, it is observed that saliency features and fingerprint specific features contribute more towards segmentation while in the IIIT-D CLF database, saliency features and image intensity based features contribute more for segmentation. Thus, across databases, saliency based features are found to primarily contribute toward segmentation along with other features that are database specific.
- For a single latent fingerprint from the NIST SD-27 database, the average feature extraction time on a Windows-7 desktop system with Intel i7 processor, $2.86GHz$ CPU, and $8GB$ RAM is $15.95ms$ while for the IIIT-D CLF database it is $10.13ms$. The computation time for extracting only the optimal features on the same desktop system is $8.07ms$ for NIST SD-27 database and $8.38ms$ for IIIT-D CLF database.

4.4. Segmentation Performance

The objective of segmentation is to extract all the foreground regions while discarding the noisy background regions. The results of the proposed segmentation algorithm are shown in Table 5 and Table 6. As segmentation is the first step in the feature extraction and matching pipeline, the ideal situation is to have high foreground segmentation accuracy (FSA), with not very low background segmentation accuracy (BSA), suggesting that the informative foreground region is minimally lost while allowing some background (noisy) regions. The segmentation accuracy along with FSA and BSA, when compared with the ground truth manual segmentation can be considered as a good estimate of the performance of a segmentation algorithm. When the ground truth of segmentation is not available, SIVV-TPR acts as a robust "as-is" metric to evaluate the performance of latent segmentation without the need for performing matching. Key observations from the segmentation results are as follows:

Table 6: The SIVV-TPR improvement on the three databases, before and after segmentation.

Algorithm	**NIST SD-4**	**NIST SD-27**	**IIIT-D CLF**
Unsegmented	0.9195 ± 0.007	0.3468 ± 0.04	0.4378 ± 0.066
Ground truth	0.9463 ± 0.005	0.5366 ± 0.033	0.5330 ± 0.014
All features + SVM	0.9267 ± 0.006	0.4726 ± 0.069	0.5092 ± 0.063
All features + Neural Network	0.9295 ± 0.009	0.4738 ± 0.047	0.5157 ± 0.045
All features + RDF	0.9325 ± 0.005	0.5168 ± 0.059	0.5512 ± 0.029
Saliency features + RDF	0.9330 ± 0.008	0.4757 ± 0.036	0.5354 ± 0.027
Optimal features + RDF	$\mathbf{0.9410 \pm 0.006}$	$\mathbf{0.5274 \pm 0.034}$	$\mathbf{0.555 \pm 0.024}$

- Table 5 shows high segmentation accuracy of about 96% on inked fingerprints from NIST SD-4 dataset. Also, the accuracy with using only the saliency features is about 78% while using the optimal features is as high as 92%. These results validate the goodness of the proposed algorithm and also the selection of optimal features.

- We analyze the performance of individual category of features on the NIST SD-27 database and observe that saliency features provide the highest foreground segmentation accuracy of 73.4% whereas the FSA of other four features is less than 30%.

- From Table 5, it is observed that for latent prints in NIST SD-27 database, the proposed RDF based algorithm (with all features) yields the best FSA of 83.41%. Also, in the IIIT-D CLF database it is observed that the proposed algorithm yields significantly higher segmentation accuracy of 93.23%. This, in general, highlights the successful adaptive nature of the proposed segmentation algorithm for different qualities of prints.

- In both the NIST SD-27 and IIIT-D CLF databases, using only optimal features yield similar segmentation performance as the complete feature set. This shows that the optimal feature set is a comprehensive representation having foreground/background distinguishing capability comparable to the entire feature set. It is interesting to note that depending on the database characteristics, optimal feature sets are different for each database but the salient features are present for all three databases. From the *No free lunch theorem*, it is well understood that the same set of features may not yield best performance across all databases. However, from the implementation perspective, there are two things to note: use of all the features yield the best accuracy and requires 15 ms per test image whereas optimal features yield similar accuracy in 7 ms. Therefore, if the training database is available for feature selection, then depending on the database characteristics, the optimal feature set can be selected and used. If the training database is not available, then all 23 features can be used for classification.

- From the results of NIST SD-27, it can be observed that using only saliency features (f_1, f_2) provides a FSA of about 73%, while addition of fingerprint specific features improves the FSA to about $83 - 85\%$. Similar improvements can be observed in IIIT-D CLF and NIST-4 databases, as well. This observation is also visually demonstrated using sample images from NIST SD-27 in Figure 11.

- The FSA of all the algorithms is comparatively higher for the IIIT-D CLF database than the NIST SD-27 database. This can be attributed to the fact that the NIST database has real world images with significant amount of background information, whereas the IIIT-D CLF database is prepared in simulated lab environment with very little background noise such as text and lines.

- From Table 6, it can be clearly observed in all three databases, that ground truth segmented images show an improved SIVV-TPR rate compared to unsegmented images. This validates the fidelity of the proposed SIVV-TPR metric and also highlights the necessity of segmentation in latent prints.

- The SIVV-TPR metric shows that in all the datasets, using optimal features with RDF classification and using all features with RDF classification provides segmentation that is comparable to the ground truth segmentation. Thus, automatic segmentation of latent prints using the proposed algorithm provides as good segmentation as ground truth, for both kind of fingerprints (inked or different qualities of latent prints).

Table 7: Segmentation accuracy (SA), foreground segmentation accuracy (FSA), and background segmentation accuracy (BSA) (in %) of the proposed and existing feature selection algorithms on the NIST SD-4, NIST SD-27, and IIIT-D combined latent fingerprint databases.

Database	Metric	MRMR	JMI	DISR	RELIEF	Modified RELIEF
NIST SD-4	FSA	89.18 ± 0.0	89.13 ± 0.0	89.16 ± 0.0	88.93 ± 0.0	92.49 ± 0.0
	BSA	82.27 ± 0.0	82.12 ± 0.0	82.37 ± 0.0	82.51 ± 0.0	91.50 ± 0.0
NIST SD-27	FSA	39.45 ± 0.0	45.33 ± 0.0	45.33 ± 0.0	43.87 ± 0.0	85.11 ± 0.1
	BSA	95.19 ± 0.0	93.76 ± 0.0	93.76 ± 0.0	94.71 ± 0.0	61.63 ± 0.2
IIIT-D CLF	FSA	96.98 ± 0.0	96.98 ± 0.0	96.97 ± 0.0	96.28 ± 0.0	93.01 ± 0.1
	BSA	85.27 ± 0.0	85.27 ± 0.0	84.37 ± 0.0	83.14 ± 0.0	93.84 ± 0.3

4.5. Comparison with Existing Feature Selection Algorithms

The modified RELIEF algorithm is compared with some popular feature selection algorithms [41] available in literature: (i) MRMR: Max-Relevance Min-Redundancy algorithm, (ii) JMI: Joint Mutual Information algorithm, (iii) DISR: Double Input Symmetrical Relevance algorithm, and (iv) RELIEF algorithm. These algorithms are individually used to select the optimal features for each of the databases. Next, an RDF classifier is trained using the optimal features selected using various feature selection algorithms.

The performance of the feature selection algorithms is compared in terms of foreground and background segmentation accuracies and is shown in Table 7. Of all the feature selection algorithms compared in Table 7, it can be observed that the proposed modified RELIEF algorithm provides a good trade-off between the overall segmentation accuracy and foreground segmentation accuracy. Further analysis into the optimal features selected by different algorithms reveal that saliency features are assigned significantly high weight in modified RELIEF. In all other algorithms, either saliency does not occur in the list of optimal features or is assigned lower weight. The advantage of the modified RELIEF feature selection algorithm is that it works better in binary classification setting with continuous features. Hence, it can be deduced that modified RELIEF algorithm is the most suitable feature selection algorithm.

4.6. Comparison with Existing Latent Print Segmentation Algorithm

The proposed segmentation algorithm is compared with existing segmentation algorithm proposed by Zhang et al. [5]. In order to compare the results and follow the experimental protocol discussed in Section 4.2, we obtained the binary masks for all the images in NIST SD-27 database from the authors of [5] and split it according to three cross validation sets. Figure 12 shows the comparison of the two algorithms in terms of segmentation accuracy on the NIST SD-27 dataset. As compared to Zhang et al. [5], the proposed algorithm gives similar background segmentation accuracy but it yields an improvement of about 18% in the overall segmentation accuracy due to high foreground accuracy.

4.7. Matching Performance

The final objective of segmenting latent fingerprints from the background is to improve the matching performance. Therefore, the performance of the proposed segmentation algorithm is also evaluated in terms of fingerprint matching accuracy after segmentation on both the latent fingerprint databases[2]. For the NIST SD-27 database, manually annotated minutiae (available along with the database) are used while minutiae for the IIIT-D CLF database are automatically extracted using VeriFinger SDK. For segmented images, only the subset minutiae lying within the segmented mask are considered for matching.

Matching latent fingerprints is a challenging research problem and there is no standard open source latent fingerprint matching SDK or commercial system freely (or low cost) available in public domain. It is observed from the literature that local Minutiae Cylinder Code (MCC) [42, 43] description for the minutiae provides state-of-the-art results [44] on latent print matching. Therefore, latent fingerprint matching results are shown using the MCC descriptors. For both the databases, images of 2000 subjects from the NIST SD-4 database [38] are appended to extend the gallery and three times random split based cross validation is performed. The performance is reported in terms of

[2]The proposed algorithm is compared with the manual ground truth of segmentation.

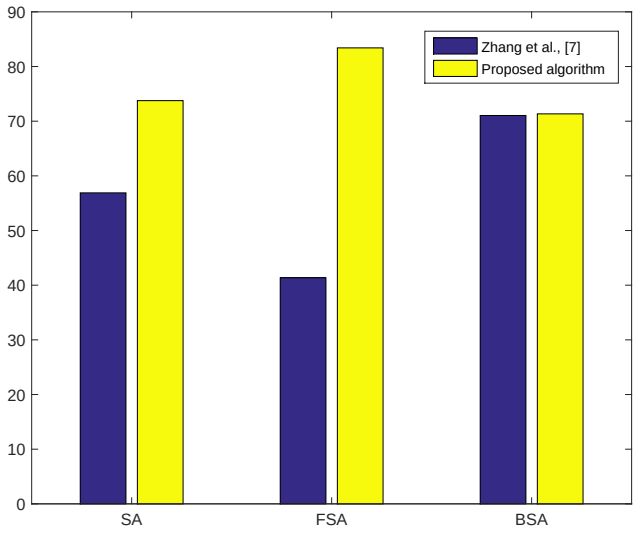

Figure 12: Comparing the performance of the proposed segmentation algorithm with the algorithm recently proposed by Zhang et al. [5], on the NIST SD-27 dataset.

Table 8: Rank-50 identification accuracy (in %) of the proposed segmentation algorithms using MCC descriptor. Manually marked minutiae are used for NIST SD-27 while VeriFinger SDK is used to extract minutiae from IIIT-D CLF database.

Algorithm	**NIST SD-27**	**IIIT-D CLF**
Unsegmented	55.9 ± 2.7	26.9 ± 2.7
Ground truth	$\mathbf{83.1 \pm 7.3}$	$\mathbf{34.2 \pm 3.6}$
All features + RDF	$\mathbf{80.0 \pm 6.3}$	$\mathbf{33.4 \pm 3.0}$
Saliency features + RDF	66.9 ± 6.5	26.6 ± 0.5
Optimal features + RDF	78.7 ± 6.4	29.6 ± 3.9

the rank-50 identification accuracy and the results are reported in Table 8 and Figures 13 and 14. Some key results obtained are as follows:

- Using ground truth segmentation, on the NIST SD-27, rank-50 identification accuracy of 83% is observed, which is significantly greater than the accuracy obtained with the unsegmented images (56%). It can be observed that both the variants of the proposed algorithm: using all features and using only optimal features, performs comparable to the ground truth segmentation with almost 80% rank-50 matching accuracy. On the NIST SD-27 database, the matching performance is not reduced much by using only the optimal set of features. Wilcoxon's rank-sum test between the results obtained from all the features and the optimal features, accepts the null hypothesis at 5% significance, claiming there is not much statistical difference between the results obtained.

- On the IIIT-D CLF database, a similar trend can be observed, where the matching performance of the proposed segmented images is almost as good as the ground truth segmented images. The performance is low due to the poor feature extraction by VeriFinger, which is fine-tuned for processing tenprints.

- The number of minutiae preserved by each algorithm after segmentation is provided in Table 9. It can be observed that, in both the databases, the percentage of minutiae preserved by the segmentation algorithm is proportional to its corresponding matching performance. A higher number of minutiae is found in the IIIT-D CLF database as they are automatically extracted using a ten-print matcher, while in NIST SD-27 manually annotated minutiae are used.

Table 9: Average number of minutiae extracted in the fingerprint images after segmentation.

Algorithm	**NIST SD-27**	**IIIT-D CLF**
Ground truth	19.2	31.9
All features + RDF	19.2	33.3
Saliency features + RDF	15.5	31.1
Optimal features + RDF	18.7	33.3

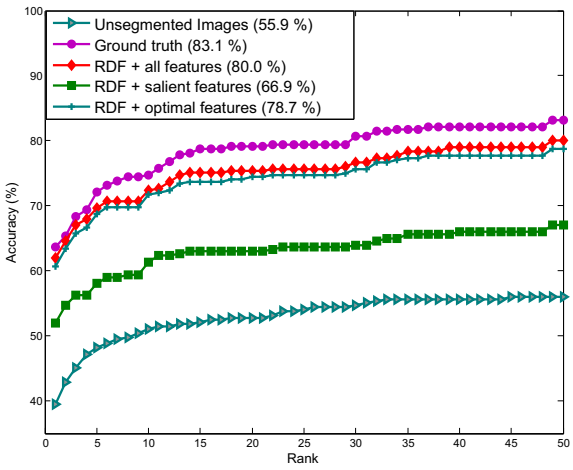

Figure 13: CMC curves showing the average matching performance of unsegmented and segmented images on the NIST SD-27 database.

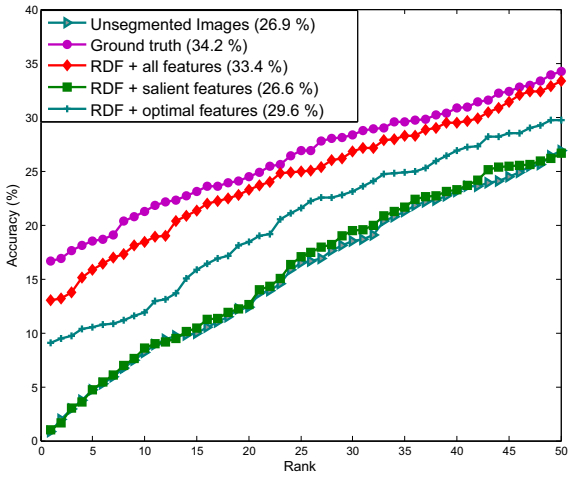

Figure 14: CMC curves showing the average matching performance of unsegmented and segmented images on the IIIT-D CLF database.

4.8. Performance Evaluation using Latent Fingerprint Identification System with Very Large Gallery

We also compute the effectiveness of the proposed segmentation algorithm using a popular latent fingerprint identification system used by law enforcement agencies[3]. The system has over 2 million pre-enrolled identities in the database and is modular in nature. Experiments are performed on the IIITD CLF database. NIST databases are not used as they may have been used to train the system apriori. First, the gallery images for IIITD CLF are enrolled (after

[3]The license agreement does not allow us to name the commercial system in any kind of comparison.

the experiments, these enrollments are deleted from the system) and then 1280 probe images are used for evaluation. Two sets of experiments are performed: (1) using the default setting of the latent fingerprint system which uses inbuilt segmentation algorithm and (ii) when segmented outputs obtained from the proposed algorithm are given as input for matching. Rank-50 accuracies obtained for both the experiments are 71.4% and 72.3% respectively. This shows that the proposed segmentation algorithms improves the identification performance of a latent fingerprint system on a very large gallery database.

5. Conclusion

Latent fingerprints collected as a forensic evidence are affected by background noise, limited content, and varying quality. As the first step in the recognition pipeline, latent fingerprint segmentation plays an important role. In this research, we proposed a novel latent print segmentation algorithm that extracts saliency, image, gradient, ridge, and quality features from local patches of the image. These features determine the characteristics of both foreground ridge and background noise. An optimal set of features are selected using modified RELIEF based feature selection algorithm and a Random Decision Forest classifier is used to learn *foreground* and *background* regions. Further, a n-degree polynomial representation of the segmented region is found to be the most optimal representation of the segmented results. The performance of the proposed algorithm is evaluated on the basis of three metrics: SIVV-TPR, segmentation accuracy (along with FSA and BSA), and rank-k identification accuracy. The results show that the proposed segmentation algorithm yields high segmentation performance on the NIST SD-4 inked print database and NIST SD-27 and IIIT-D CLF latent databases, showing that the algorithm is able to segment the regions of interest from the background. Using the automatically segmented images, we have observed improved matching performance, which further supports the effectiveness of the segmentation algorithm.

6. Acknowledgement

Mayank Vatsa was partly supported through the DST FAST grant from the Department of Science and Technology, India. Anush Sankaran is partly supported by the TCS PhD research fellowship. The authors acknowledge Prof. C.-C. Jay Kuo and Dr. Jiangyang Zhang for sharing the fingerprint segmentation masks generated from their algorithm [5]. The authors also thank Dr. Keith Morris (West Virginia University) for his insightful discussions and support on latent fingerprint.

7. References

[1] FBI, "Next Generation Identification", http://www.fbi.gov/about-us/cjis/fingerprints_biometrics/ngi.
[2] NIST Special Database-27, http://www.nist.gov/itl/iad/ig/sd27a.cfm.
[3] H. Guan, A. Dienstfrey, M. Theofanos, B. Stanton, A Measurement Metric for Forensic Latent Fingerprint Preprocessing, 2014.
[4] A. Sankaran, M. Vatsa, R. Singh, Latent fingerprint matching: A survey, IEEE Access 2 (2014) 982–1004.
[5] J. Zhang, R. Lai, C. Kuo, Adaptive directional total-variation model for latent fingerprint segmentation, IEEE Transactions on Information Forensics and Security 8 (8) (2013) 1261–1273.
[6] K. Cao, E. Liu, A. Jain, Segmentation and enhancement of latent fingerprints: A coarse to fine ridge structure dictionary, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (9) (2014) 1847–1859.
[7] S. Karimi-Ashtiani, C.-C. Kuo, A robust technique for latent fingerprint image segmentation and enhancement, in: Proceedings of International Conference on Image Processing, 2008, pp. 1492 –1495.
[8] N. Short, M. Hsiao, A. Abbott, E. Fox, Latent fingerprint segmentation using ridge template correlation, in: Proceedings of International Conference on Imaging for Crime Detection and Prevention, 2011, pp. 1–6.
[9] H.-S. Choi, M. Boaventura, I. A. G. Boaventura, A. K. Jain, Automatic segmentation of latent fingerprints, in: Proceedings of IEEE International Conference on Biometrics: Theory Applications and Systems, 2012.
[10] NBIS (NIST Biometric Image Software), Developed by National Institute of Standards and Technology (NIST), http://www.nist.gov/itl/iad/ig/nbis.cfm.
[11] A. M. Bazen, S. H. Gerez, Segmentation of fingerprint images, in: Workshop on Circuits Systems and Signal Processing, 2001, pp. 276–280.
[12] D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, Handbook of fingerprint recognition, Springer-Verlag.
[13] NIST, NFIQ 2.0, Quality features definition, http://biometrics.nist.gov/cs_links/quality/NFIQ_2/NFIQ-2_Quality_Feature_Defin-Ver05.pdf.
[14] E. Zhu, J. Yin, C. Hu, G. Zhang, A systematic method for fingerprint ridge orientation estimation and image segmentation, Pattern Recognition 39 (8) (2006) 1452–1472.

[15] A. Sankaran, T. I. Dhamecha, M. Vatsa, R. Singh, On matching latent to latent fingerprints, in: Proceedings of International Joint Conference on Biometrics, 2011, pp. 1–6.
[16] A. Sankaran, M. Vatsa, R. Singh, Hierarchical fusion for matching simultaneous latent fingerprint, in: Proceedings of International Conference on Biometrics: Theory, Applications and Systems, 2012, pp. 1–6.
[17] A. Borji, What is a salient object? a dataset and a baseline model for salient object detection, IEEE Transactions on Image Processing 24 (2) (2015) 742–756.
[18] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: Proceedings of Advances in Neural Information Processing Systems, 2006, pp. 545–552.
[19] X. Chen, J. Tian, J. Cheng, X. Yang, Segmentation of fingerprint images using linear classifier, EURASIP Journal on Advances in Signal Processing 2004 (4) (2004) 480–494.
[20] C. Gottschlich, Curved-region-based ridge frequency estimation and curved gabor filters for fingerprint image enhancement, IEEE Transactions on Image Processing 21 (4) (2012) 2220–2227.
[21] S. Chikkerur, V. Govindaraju, A. N. Cartwright, Fingerprint image enhancement using STFT analysis, Pattern Recognition and Image Analysis 3687 (2005) 20–29.
[22] H. Fronthaler, K. Kollreider, J. Bigun, Local features for enhancement and minutiae extraction in fingerprints, IEEE Transactions on Image Processing 17 (3) (2008) 354–363.
[23] J. S. Bartunek, M. Nilsson, B. Sallberg, I. Claesson, Adaptive fingerprint image enhancement with emphasis on preprocessing of data, IEEE Transactions on Image Processing 22 (2) (2013) 644–656.
[24] Y. Chin, T. Ong, A. Teoh, K. Goh, Integrated biometrics template protection technique based on fingerprint and palmprint feature-level fusion, Information Fusion 18 (2014) 161–174.
[25] M. B. Salah, A. Mitiche, I. B. Ayed, Multiregion image segmentation by parametric kernel graph cuts, IEEE Transactions on Image Processing 20 (2) (2011) 545–557.
[26] S. Yoon, E. Liu, A. K. Jain, On latent fingerprint image quality, in: Proceedings of International Workshop on Computational Forensics, 2012.
[27] A. Tsymbal, S. Puuronen, D. W. Patterson, Ensemble feature selection with the simple bayesian classification, Information Fusion 4 (2) (2003) 87–100.
[28] A. Tsymbal, M. Pechenizkiy, P. Cunningham, Diversity in search strategies for ensemble feature selection, Information fusion 6 (1) (2005) 83–98.
[29] K. Kira, L. A. Rendell, The feature selection problem: Traditional methods and a new algorithm, in: AAAI, 1992, pp. 129–134.
[30] M. Robnik-Šikonja, I. Kononenko, Theoretical and empirical analysis of relieff and rrelieff, Machine learning 53 (1-2) (2003) 23–69.
[31] T. K. Ho, The random subspace method for constructing decision forests, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 832–844.
[32] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.
[33] L. Rokach, Decision forest: Twenty years of research, Information Fusion 27 (2016) 111–125.
[34] M. A. Olsen, H. Xu, C. Busch, Gabor filters as candidate quality measure for NFIQ 2.0, in: International Conference on Biometrics, IEEE, 2012, pp. 158–163.
[35] R. M. Haralock, L. G. Shapiro, Computer and robot vision, Addison-Wesley Longman Publishing Co., Inc., 1991.
[36] H. Guan, A. Dienstfrey, M. Theofanos, A new metric for latent fingerprint image preprocessing, in: Computer Vision and Pattern Recognition Workshops, 2013, pp. 84–91.
[37] J. M. Libert, J. Grantham, S. Orandi, A 1D Spectral Image Validation/Verification Metric for Fingerprints, 2009.
[38] Fingerprint minutiae from latent and matching tenprint images, NIST Special Database 4.
[39] B. Ulery, R. Hicklin, J. Buscaglia, M. Roberts, Accuracy and reliability of forensic latent fingerprint decisions, Proceedings of the National Academy of Sciences 108 (19) (2011) 7733–7738.
[40] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.
[41] G. Brown, A. Pocock, M.-J. Zhao, M. Luján, Conditional likelihood maximisation: A unifying framework for information theoretic feature selection, The Journal of Machine Learning Research 13 (1) (2012) 27–66.
[42] R. Cappelli, M. Ferrara, D. Maltoni, Minutia cylinder-code: A new representation and matching technique for fingerprint recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (12) (2010) 2128–2141.
[43] M. Ferrara, D. Maltoni, R. Cappelli, Noninvertible minutia cylinder-code representation, IEEE Transactions on Information Forensics and Security 7 (6) (2012) 1727–1737.
[44] A. A. Paulino, J. Feng, A. K. Jain, Latent fingerprint matching using descriptor-based hough transform, IEEE Transactions on Information Forensics and Security 8 (1) (2013) 31–45.