Journal of Scientific & Industrial Research
Vol. 79, August 2020, pp. 730-735

Latent Fingerprint Indexing for Faster Retrieval from Dataset with Image Enhancement Technique

Harivans Pratap Singh[1,3], Priti Dimri[2], Shailesh Tiwari[3]*, and Manish Saraswat[3]

[1]Department of Computer Science & Engineering, UTU Dehradun, Uttarakhand, India

[2]Department of Computer Science and Applications, G B Pant Engineering College, Ghurdauri, Uttarakhand, India

[3]ABES Engineering College, Ghaziabad, UP, India,

Received 09 August 2019; revised 29 March 2020; accepted 19 April 2020

Since decades fingerprints have been the prime source in identification of suspects latent fingerprints are compared and examined with rolled and plain fingerprints which are stored in the dataset. The common challenges which are faced while examining latent fingerprints are background noise, nonlinear distortions, poor ridge clarity and partial impression of the finger. As conventional methods of Segmentation doesn't perform well on latent fingerprints. The current advancement in machine learning based segmentation approach has been showing good results in terms of segmentation accuracy but lacks to provide accurate result in terms of matching accuracy. As one of the problem faced in matching latent fingerprint is low clarity of ridge-valley pattern which results in detection of false minutiae and poor matching accuracy. A multilayer processing of artificial neural network based segmentation is proposed to minimize the detection of false minutiae and increase the matching accuracy. This approach is designed on binary classification model where the simulation will be carried out on IIIT-D latent fingerprint dataset. Segmentation will be divided into full and partial impression fingerprints which are then compared with minutiae with the database using local and global matching algorithm. An improvised result is received which is more accurate as compared to the previous algorithms.

Keywords: Latent fingerprints, Minutiae, Multi-layer Neural network, Segmentation

Introduction

Latent fingerprints are extracted from rolled and plain surfaces. Comparisons are made between latent fingerprint[1,2] and templates and as a result templates having minimum similarity score are considered as best match. Rolled and plain fingerprints are extracted by busing live-scan fingerprint scanners like optical or capacitive scanners these fingerprints thus obtained are considered as good quality fingerprints.[3,4] Also offline fingerprint capturing such as use of ink are good methods, under expert supervision. The process involved is highly complex chemical process to lift the latent fingerprints.

Latent fingerprints comparatively have smaller number of minutiae as compared to exemplars. Mostly exemplars have average 106 number of minutiae against 21 minutiae on National Institute of Standards and Technology (NIST) Special latent Database (SD27)[2] for latent fingerprints[5,8] any decrease in number of minutiae could severely affect the matching accuracy. National Institute of Standards and Technology has worked on 10,000 exemplars and it has shown the best score of 99.4% hence there is not much research space available.[5]

*Author for Correspondence
E-mail: shail.tiwari@yahoo.com

Experimental Details

Latent Fingerprint Image Quality Enhancement

Quality of the image has a great impact on the performance of a fingerprint recognition system. The goal of image enhancement is to improve the overall performance of the input image; it improves the clarity of ridge valley structure of fingerprints which result in minimizing the false detection of minutiae.[6] Image enhancements possess three important characteristics when dealing with fingerprints:

1. Broken edge reconnection which may be caused by scars or dryness of finger
2. Preservation of bifurcation and ridge endings
3. Separation of false conglutinated ridges which may be caused by smudges or wetness of finger

In this paper first the image is divided into blocks of size $w \times w$ and then each block is filtered with Gabor Filter (GF). Gabor Filter is a 2-D filter which is formed by the combination of a cosine with a 2-D Gaussian function. General form of GF is:

$$GF(x, y, \theta, f, \sigma_x, \sigma_y) = \exp\left\{-\frac{1}{2}\left[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right]\right\}.\cos(2\pi f x_\theta) \quad \ldots (1)$$

Where,

$$x_\theta = x.\cos\theta + y.sin\theta \quad \ldots (2)$$
$$y_\theta = -x.\sin\theta + y.cos\theta \quad \ldots (3)$$

By Eq. (1) it is represented that the Gabor Filter is centered at origin, rotation of this filter around x and y axis is represented by θ, f represents local frequency, and σ_x, σ_y are the standard deviation of the Gaussian function along the x and y axis.

Segmentation

In this section segmentation approach is discussed in detail. For segmentation block wise processing is also followed

Features Extraction

Since the objective of this segmentation model is to categorize local block of size $w \times w$ into foreground or fingerprint region and background or non-fingerprint region. For this model to work efficiently, we have extracted fingerprint's specific features such as features based on image intensity, gradient, and ridge

Gradient based Features

Gradient measures the directional change in pixel intensity, therefore fingerprint region would have more regular directional change as compared to non-fingerprint region. It is also used to measure the orientation of ridges in a local block and a good feature to differentiate non-fingerprint pattern or background noise from the fingerprint pattern. Orientation at a point (p, q)can be calculated as:

$$O(p,q) = \begin{cases} \pi/4, & a = 0, b < 0 \\ 3\pi/4, & a = 0, b \geq 0 \\ \theta(p,q) + \pi/2, & a > 0 \\ \theta(p,q), & a < 0, b \leq 0 \\ \theta(p,q) + \pi, & a < 0, b > 0 \end{cases} \quad \ldots (4)$$

Where, $\theta(p,q)$, a, and b are defined as follows:

$$\theta(p,q) = \frac{1}{2}\tan^{-1}\left(\frac{b}{a}\right) \quad \ldots (5)$$

$$a = \sum_{p=1}^{w}\sum_{q=1}^{w}(I_x^2(p,q) - I_y^2(p,q)) \quad \ldots (6)$$
$$b = \sum_{p=1}^{w}\sum_{q=1}^{w} 2 * I_x(p,q) * I_y(p,q) \quad \ldots (7)$$

$I_x, I_y$are the gradient along x and y direction respectively.

Features based on gradient with the help of Eqs (4–7) can be calculated as:

(i) Ridge orientation: It can be calculated using Gaussian smoothing kernel.[7]

$$F1 = \frac{1}{w^2}\sum_{p=1}^{w}\sum_{q=1}^{w} O'(p,q) \quad \ldots (8)$$

$$O'(p,q) = \frac{1}{2}\tan^{-1}\left(\frac{\sin(2O(p,q)) * G(p,q)}{\cos(2O(p,q)) * G(p,q)}\right) \quad \ldots (9)$$

Where, $G(p,q)$ is the Gaussian smoothing kernel.

(ii) Gradient sum square: Interleaving ridge-valley pattern provides a good measure of change in flow and it is maximum for fingerprint region. Formula to calculate this interleaving ridge-valley pattern as a square of sum of gradient is shown in Eq. (10).

$$F2 = \sqrt{a^2 + b^2} \quad \ldots (10)$$

(iii) Sum of norm of squared gradient:

$$F3 = \sum_{p=1}^{w}\sum_{q=1}^{w}(I_x^2(p,q) - I_y^2(p,q))^2 + (2 * I_x(p,q) * I_y(p,q))^2 \quad \ldots (11)$$

Ridge based Features

Ridge based features have a very good measure to differentiate between latent fingerprint from the noisy or other fingerprints in the background. We have obtained four different features based on ridge.

(i) Ridge frequency: Fourier transformation is applied to each local block of size $w \times w$ to obtain this feature. It can be calculated using Eq. (12)

$$F4 = argmax(\sum_{p=1}^{w}\sum_{q=1}^{w}|X(p,q)| * F_n(p,q)) \quad \ldots (12)$$

Where, $|X(p,q)|$ represents the Fourier transformation of local image block and $F_n(p,q)$ represents the n^{th} directional filter. Frequency of the filter gives maximum response is considered as ridge frequency.

(ii) Inter-ridge average distance: The inter-ridge average distance supports in differentiating the fingerprint region with non-fingerprint region. As fingerprint region have higher number of ridges, therefore this inter-ridge average distance would be minimum as compared to non-fingerprint region. It can be calculated using Eq. (13) as:

$$F5 = \frac{\sum_{d=1}^{M} D_d}{P-1} \quad \ldots (13)$$

Where, M is the number of ridges peaks and D_d is the consecutive peak distance.

(iii) Peak heights ridges variance:

Variance in ridge pressure in $w \times w$ block size can be computed using Eq. (14)

$$F6 = \frac{\sum_{\square=1}^{M}(P_l - P_{mean})}{M-1} \quad \ldots (14)$$

Where, $P_{\square}$ is the ridge height of $\square^{t\square}$ peak and P_{mean} is the average ridge height of all peaks across all blocks.

(iv) Ridge Energy: Ridgeness of the local block of size $w \times w$ is expected to be more in fingerprint region. It is the measure of confidence of local blocks and is very helpful for the segmentation of latent fingerprint. It is also called as ridge energy and can be calculated using Eq. (15) as follows:

$$F7 = \frac{1}{w^2}\left(\sum_{p=1}^{w}\sum_{q=1}^{w}(|X(p,q)| * F_n(p,q))^2\right) \quad \ldots (15)$$

Image Intensity based Features:

Pixel and its intensity are the basic building block of any image. Like gradient and ridge based features, image intensity based features are also fingerprint specific as fingerprint region have regular pattern than non-fingerprint region. Therefore, the intensity related features for fingerprint regions have a large difference than their counter part as shown in below Fig.1 (a), (b) & (c). We have calculated three different features based on intensity in order to support classifier to better classify the fingerprint region. These features are:

(i) Local and global mean difference: This feature is the measure of how far the fingerprint regions are from the global mean or average grayscale value. The fingerprint patterns are regular as compared to noisy background as shown in Fig. 2. Therefore, this difference would be minimum for fingerprint regions. This feature can be obtained using Eq. (16): $F8 =$

$$\left(\frac{1}{w^2}\sum_{p=1}^{w}\sum_{q=1}^{w} I(p,q)\right) - I_{mean} \quad \ldots (16)$$

Where, I_{mean} is the average intensity of complete image and $I(p,q)$ is the intensity at pixel location(p,q).

(ii) Local variance:

The variation in intensities in local blocks can be captured through this feature. Due to the presence of interleaved ridge-valley structure, the variance would be more in fingerprint region. It can be calculated using Eq. (17) as:

$$F9\, \frac{1}{w^2}\sum_{p=1}^{w}\sum_{q=1}^{w}\left(I(p,q) - \frac{1}{w^2}\sum_{p=1}^{w}\sum_{q=1}^{w} I(p,q)\right)^2 \quad \ldots (17)$$

Fingerprint patterns or foreground ridge blocks are shown in (a) and noisy background in (b) of Fig. 3.

(iii) Local ridge pixels clustering: The ridge valley structure present in the fingerprint can be captured with the help of local ridge pixels clustering. It combines the properties of mean and variance of

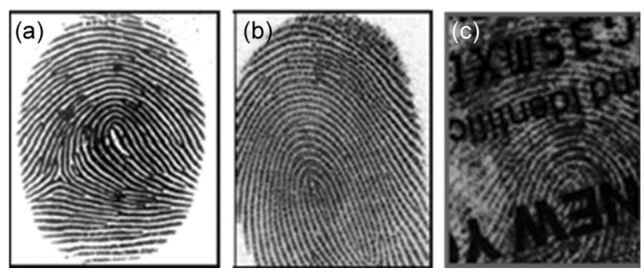

Fig. 1 — (a) Live scan fingerprint (b) Inked fingerprint and (c) Latent fingerprint

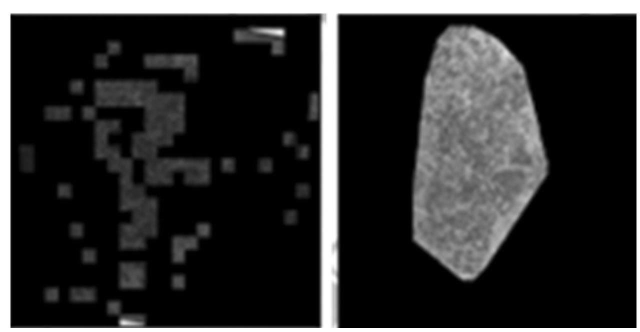

Fig. 2 — Segmented local blocks and the final image in the region of interest (ROI)

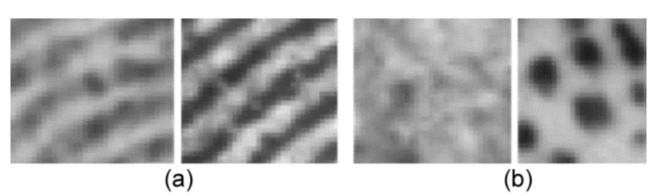

Fig. 3 — Fingerprint patterns or foreground ridge blocks in (a) and noisy background in (b)

image intensity of local blocks to create this feature. This feature can be computed using Eq. (18)

$$F10 = \sum_{p=1}^{w} \sum_{q=1}^{w} u1(p,q) \times u2(p,q) \quad \dots (18)$$

Where

$$u_1(p,q) = \begin{cases} 1 & if I(p,q) < I_{mean} \\ 0 & ot\square erwise \end{cases}$$

$$u_2(p,q) = \begin{cases} 1\ if F(p,q) < \left(\dfrac{w^2}{2} + 1\right) \\ 0\ ot\square erwise. \end{cases}$$

$$F(p,q) = \sum_{l=p-\frac{w}{2}}^{l=p+\frac{w}{2}} \sum_{m=q-\frac{w}{2}}^{m=q+\frac{w}{2}} u_1(l,m)$$

Here, F is the degree of uniformity in a local block of size $w \times w$ and it tends to be larger in uniform background regions than in ridge valley regions.

Features vector of the local block of size $w \times w$ would be $[F1, F2, F3, F4, F5, F6, F7, F8, F9, F10]$.

Classification using neural network

Generally, the real data are not linearly separable, therefore FCMLFFNN by varying the window size is considered as a best suited model in this paper to classify each block into foreground (fingerprint region). Region of Interest (ROI) can be obtained by combining the blocks at the edge.[8]

Multilayer Feed Forward Neural Network

The term feed forward means one layer of neurons feeds forward to the next layer of neurons and so on. All the neurons at one layer are fully connected with the neurons at another layer and so on, hence it is also known as fully connected MLFFNN (FCMLFFNN). Number of neurons at each layer as well as number of hidden layers is application dependent. Each connection between nodes has a weight associated with it. There are special weights (w0 and z0) that feed into every node at hidden layer and output layer. These special weights are called bias and set thresholding values for the nodes. Initially, all weights are set to very small random values near to zero and these weights get updated during training.

Network Description

Input units at input layer are application dependent. No processing performs at input layer and all the inputs provided at input layer are feed into the system for processing. Every node (n_k) is connected with all nodes at hidden units and each connection is associated with a ($w_{\square k}$).

In a network there could be one or more hidden layers. All the input units are connected with the hidden units at hidden layer. Each hidden node calculates the weighted sum of all inputs and applies a thresholding to determine the output of that hidden unit. Weighted sum can be calculated as:

$$\sum_{k=0}^{D} w_{\square k} * x_k \quad \dots (19)$$

Sigmoidal function is applied at each hidden unit to calculate the output. Sigmoidal function is represented as:

$$sigmoid(x) = \frac{1}{1+e^{-x}} \quad \dots (20)$$

Sigmoidal function squashes the given input x into range 0 to 1. Therefore, sigmoidal of weighted sum of inputs is:

$$0 \le S_{\square} = \frac{1}{1+e^{-\sum_{k=0}^{D} w_{\square k} * x_k}} \le 1 \quad \dots (21)$$

Where $\square \in [0, N]$, and N total number of hidden units

Training Network

Training neural network for a given set of inputs is an iterative process in which the network is trained using back propagation algorithm by minimizing the error between the actual outputs given by the network and the desired (target) outputs. The general idea of back propagation algorithm is to update weights in each iteration by using gradient descent algorithm to minimize the error. The weights are updated by taking the partial derivative of the error function with respect to the weights to determine each weight's contribution to the error.

Error of output unit i for training example (x^i, t^i) is:

$$E\left(w, V \middle| x^i, t^i\right) = \frac{1}{2}\left(t^i - y^i\right)^2 \quad \dots (22)$$

Where, $E\left(w, V \middle| x^i, t^i\right)$ is the error for the set of weights w, V and for given training data (x^i, t^i). Error for all output units is simply the sum of error overall output units, as follows:

$$E\left(w, V \middle| x^i, t^i\right) = \frac{1}{2}\sum_{i=1}^{T}\left(t^i - y^i\right)^2 \quad \dots (23)$$

Where, T is the total number of output units

Results and Discussion

In this paper we have used latent fingerprint dataset of IIIT-D. The latent fingerprint contains 15 subjects with all 10 fingerprints but due to the uniqueness of the finger hence we have total of 150 categories.

Backgrounds have been considered into two forms: ceramic plate and tile. Multiple session wise dataset is being prepared so that it can identify factors like dryness, wetness and moisture lifting of fingerprints. It is done by using chemical process which has been captured using Canon EOS 500D with 15 Mega pixels camera having resolution (4752 × 3168). These 150 exemplars and 1046 latent fingerprints corresponding to 150 categories. These have been scanned at 1000 dpi, as depicted in the Fig. 3, some showing single impression whereas some having multiple impressions.

Overlapped images as well as partial impressions are enhanced using GF segmentation by applying separately to foreground and background block. A supervised learning approach is developed in a training set of 1000 positive and 1000 negative samples of size $w \times w$. The positive samples constitute normal fingerprint as well as latent fingerprint images, while negative samples contain non-fingerprints. Therefore two different window size such as 16 × 16 and 32 × 32 are considered for the development of training sets. First, FCMLFFNN has been trained on training sets then the trained model is tested on latent fingerprints. The proposed segmentation approach segments total 2078 fingerprints from total 1046 latent fingerprint images. These segmented images are further divided into full impression and partial impression fingerprints. In total we have 1600 full impression and 478 partial impression fingerprints. The fingerprints having partial impression of the finger, relatively smaller friction ridge pattern area, and poor quality of ridge due to background noise are called partial impression fingerprints. All the simulations are performed in MATLAB 2018a. Results in terms of segmentation accuracy are calculated for before and after enhancement.

Results of the proposed approach are compared with results obtained through support vector machine (SVM). Since, SVM is a linear classifier; therefore non-linear kernels (Gaussian and Radial Basis Function (RBF)) are used to train the SVM model. Bayesian optimization is used for SVM's hyper parameters optimization during training. Parameters used in the image enhancement approach are: quadratic window of size 11 × 11 pixels, and the standard deviation for Gaussian function are $\sigma_x = \sigma_y = 4.0$. The comparison results of segmentation accuracy for different cases are shown in Table 1. Segmentation accuracy is calculated by comparing the segmented results with manually segmented ground truth image. Results show that the average segmentation accuracy of the proposed approach has been increased by 4.15% with 32 × 32 window size as compared to the segmentation accuracy obtained without using image enhancement technique. Also proposed approach outperforms SVM in all cases. In both the cases, SVM with Gaussian kernel outperforms RBF kernel whereas, Gaussian kernel with 32 × 32 window size has better performance than their counterpart.

The efficacy of segmentation algorithm can be observed through matching accuracy. As discussed above, we have total 150 exemplars of size 661 × 508, while the sizes of segmented latent fingerprints are not certain. For this reason we have used a minutiae matching based on local and global structure. The major advantage of this matching algorithm is that it is independent of rotational, translational, and size factors of query image. Proposed approach as well as SVM yields good results with 32 × 32 window size whereas, SVM with Gaussian kernel produces better

Table 1 — Comparison of segmentation accuracy

Classifier	Kernel	Window Size ($w \times w$)	Segmentation Accuracy (in %)	
			Only Segmentation	Segmentation with image enhancement
SVM	RBF	16x16	80.25	84.48
SVM	RBF	32x32	83.08	86.14
SVM	Gaussian	16x16	87.32	90.93
SVM	Gaussian	32x32	91.11	94.19
FCMLFFNN	—	16x16	92.83	96.87
FCMLFFNN	—	32x32	94.77	98.92

Table 2 — Comparison of matching accuracy

Number of Images	Accuracy (in %)			
	Only Segmentation		Segmentation with image enhancement	
	SVM	FCMLFFNN	SVM	FCMLFFNN
1600 full impression fingerprints	79.25	94.63	84.30	97.20
478 partial impression fingerprints	55.95	71.33	63.38	83.20

segmentation accuracy. Algorithm discussed in Altuntas *et al*. (2018)[2] has used the calculation of the minutiae. Minutiae extracted from first three subjects are shown in Inverted skeleton image with core point (green), bifurcation (blue for $\theta \in [0°, 180°)$ and purple for $\theta \in [180°, 360°)$, delta points (gold), and ridge endings (orange for $\theta \in [0°, 180°)$ and red for $\theta \in [180°, 360°)$.

As the exemplar images are of high quality, therefore minimum, average, and maximum numbers of minutiae are 65, 102, and 150 which is quite high as compared to the minimum required minutiae. Accordingly, if a fingerprint has more than 12 minutiae then identity is certain. Therefore, we have removed all the minutiae present around the corners as it has very little or no impact on accuracy. Even after removal, average number of minutiae is 90. Hence, we have two types of segmented latent fingerprint images: (i) full and (ii) partial. Average numbers of minutiae in full latent fingerprints images are 70 while average numbers of minutiae in partial latent fingerprints are 24. Therefore, matching accuracy of full and partial latent fingerprints are calculated separately. Matching accuracy is calculated for both the segmentation only and the proposed approach is shown in Table 2. On full impression fingerprints matching accuracy is 98.30% which is quite high and 3% more than only segmentation approach. The major improvement can be observed in partial impression fingerprints as the accuracy has been reached up to 84.19% from 72.34%. Since, the average numbers of minutiae in partial impression fingerprints are 24, therefore the accuracy is poor with only segmentation algorithm.

Conclusions

This paper discusses the latent fingerprint indexing approach for faster retrieval of fingerprint from the available dataset with image enhancement technique. It has used the multi-layer artificial neural network approach to handle poor matching accuracy. Through Gabor filter and using image enhancement technique for segmentation, linear classification model is designed to improve the ridge valley clarity as a result the false minutiae detection is minimized. In correlation with the multi-layer feed forward neural network classification model is design which shows outstanding results in identification of latent fingerprint indexing with the help of latent fingerprint enhancement and segmentation techniques. This study proposes, estimation of fingerprint retrieval and the correctness of orientation by machine learning using neural network. The correctness is responded by the trained neural network to a block orientation which indicates the quality of the block. The estimated orientations are for correcting falsely segmentations.

References

1 Sankaran A, Jain A, Vashisth T, Vatsa M & Singh R, Adaptive latent fingerprint segmentation using feature selection and random decision forest classification, *Int J Fusion*, **34** (2017) 1–15.

2 Altuntas V, Altuntas S & Gok M, Automatic colony segmentation on agar surface by image processing, *J Sci Ind Res*, **77(8)** (2018) 451–456.

3 Liu M, Liu S, Yan W, Latent fingerprint segmentation based on ridge density and orientation consistency, *Secur Commun Netw* (2018) 1–10.

4 Khodadoust J, Khodadoust A M, Fingerprint indexing based on minutiae pairs and convex core point, *Pattern Recognit*, **67** (2017) 110–126.

5 Singh H P & Dimri P, A survey of latent fingerprint indexing and segmentation based matching, in *Advances in Data and Information Sciences* by M Kolhe, S Tiwari, M Trivedi, K Mishra, Lecture Notes in Networks and Systems, Vol 94, Springer, Singapore, 2020, 677–685, https://doi.org/10.1007/978-981-15-0694-9_62.

6 Singh H P, Dimri P & Tiwari S, Latent fingerprint indexing and segmentation techniques, *Int J Innov Technol Explor Engr*, 8**(6)** (**2019**) 1234–1238.

7 Singh H P, Segmentation of latent fingerprint using neural network, *Int J Engr Adv Technol*, **9(1)** (2019) 3777–3780.

8 Liu S, Liu M, & Yang Z, Latent fingerprint segmentation based on linear density, *proc International Conference on Biometrics (ICB)*, Halmstad, 2016, p. 1–6, doi: 10.1109/ICB.2016.7550076.