IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
ISSN: 2278-2834, ISBN: 2278-8735, PP: 13-18
www.iosrjournals.org

Finger Print Image Enhancement and Point Pattern Based Verification

[1]Amol R. Yadav, [2]Mrs K.V.Kulhalli
(D. Y. Patil college of Engg. & Tachnology Kolhapur, Maharashtra,India)
(D. Y. Patil college of Engg. & Tachnology Kolhapur, Maharashtra,India

Abstract
Fingerprints are the oldest and most widely used form of biometric identification. Despite the widespread use of fingerprints, there is little statistical theory on the uniqueness of fingerprint minutiae. A critical step in studying the statistics of fingerprint minutiae is to reliably extract minutiae from the fingerprint images. However, fingerprint images are rarely of effect quality. They may be degraded and corrupted due to variations in skin and impression conditions. Thus, image enhancement techniques are employed prior to minutiae extraction to obtain a more reliable estimation of minutiae locations for verification. This paper presents finger print image enhancement process. And verification by using minutia points.

Why for image enhancement?
There are a lot of complicating factors in minutiae matching.
1. Both sets may suffer from false, missed and displaced minutiae, caused by imperfections in the minutiae extraction stage.
2. The two fingerprints to be compared may originate from a different part of the same finger, which means that both sets overlap only partially.
3. The two prints may be translated, rotated and scaled with respect to each other.
4. The fourth problem is the presence of non-linear plastic distortions or elastic deformations in the fingerprints, which is the most difficult problem to solve. These distortions are caused by the acquisition process itself. During capturing, the 3-dimensional elastic surface of a finger is pressed on a flat sensor surface. This 3D-to-2D mapping of the finger skin introduces non-linear distortions, especially when forces are applied that are not orthogonal to the sensor surface. The effect is that the sets of minutiae of two prints of the same finger no longer fit exactly after rigid registration. This is illustrated in Fig.

Rigid registration
Fig .1

TPS registration
fig. 2

Image enhancement algorithm:
It consists the following stages:
1 Segmentation,
2 Normalization,
3 Orientation estimation,
4 Ridge frequency estimation,
5 Gabor filtering,
6 Binarisation,
7 Thinning.

1. Segmentation:

The first step of the fingerprint enhancement algorithm is image segmentation. Segmentation is the process of separating the foreground regions in the image. From the background regions. The foreground regions correspond to the clear fingerprint area containing the ridges and valleys, which is the area of interest.

The background corresponds to the regions outside the borders of the fingerprint area, which do not contain any valid fingerprint information

Steps for segmentation:

1. Firstly, the image is divided into blocks.
2. then grey-scale variance is calculated for each block in the image.
3. If the variance is less than the global threshold, then the block is assigned to be a background region; otherwise, it is assigned to be part of the foreground.
4. The grey-level variance for a block of size *W x W* is defined as:

$$V(k) = \frac{1}{W^2} \sum_{i=0}^{W-1} \sum_{j=0}^{W-1} (I(i,j) - M(k))^2$$

Where *V* (*k*) is the variance for block *k*, *I*(*i; j*) is the grey-level value at pixel (*i; j*), and *M*(*k*) is the mean grey-level value for the block *k*.

2. Normalisation:

The next step in the fingerprint enhancement process is image normalization.
Normalisation is used to standardize the intensity values in an image by adjusting the range of grey-level values so that it lies within a desired range of values. Let *I*(*i; j*) represent the grey-level value at pixel (*i; j*), and *N*(*i; j*) represent the
normalised grey-level value at pixel (*i; j*). The normalized image is defined as:

$$N(i,j) = \begin{cases} M_0 + \sqrt{\frac{V_0(I(i,j)-M)^2}{V}} & \text{if } I(i,j) > M, \\ M_0 - \sqrt{\frac{V_0(I(i,j)-M)^2}{V}} & \text{otherwise,} \end{cases}$$

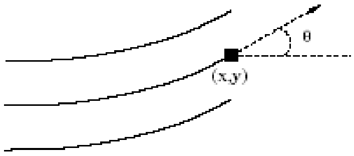
Where *M* and *V* are the estimated mean and variance of *I*(*i; j*), respectively, and M0 and V0 are the desired mean and variance values, respectively.

fig.3

fig.4

3. The orientation field of a fingerprint image:

The orientation field of a fingerprint image defines the local orientation of the ridges contained in the fingerprint. The orientation estimation is a fundamental step in the enhancement process as the subsequent Gabor filtering stage relies on the local orientation in order to effectively enhance the fingerprint image.



The orientation of a ridge pixel in a fingerprint.

Orientation field estimation algorithm steps:

1. Divide the input fingerprint image into non-overlapping blocks of size W x W
2. Compute the gradients Gx (i, j) and Gy (i, j) at each pixel (i, j) using Sobel operator. The horizontal Sobel operator is used to compute $\partial x(i; j)$,The vertical Sobel operator is used to compute $\partial y(i; j)$
3. Least squares estimate of the local orientation of the block centered at (i, j) is
4. Smooth the orientation field in a local neighborhood using a Gaussian filter. The orientation image is firstly converted into a continuous vector field, which is defined as:

$$\Phi_x(i,j) = cos(2\theta(i,j)),$$

$$\Phi_y(i,j) = sin(2\theta(i,j)),$$

$$\Phi'_x(i,j) = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} h(u,v)\Phi_x(i-uw, j-vw) \text{ and}$$

$$\Phi'_y(i,j) = \sum_{u=-w_\Phi/2}^{w_\Phi/2} \sum_{v=-w_\Phi/2}^{w_\Phi/2} h(u,v)\Phi_y(i-uw, j-vw),$$

5. The final smoothed orientation field *O* at pixel (*i;j*) is defined as:

$$O(i,j) = \frac{1}{2} tan^{-1} \frac{\Phi'_y(i,j)}{\Phi'_x(i,j)}.$$

4. Local ridge frequency:

This involves smoothing the projected waveform using a Gaussian low pass filter of size *w x w* to reduce the effect of noise in the projection. The ridge spacing $S(i; j)$ is then computed by counting the median number of pixels between consecutive minima points in the projected waveform. Hence, the ridge frequency $F(i; j)$ for a block centered at pixel $(i; j)$ is defined as:

$F(i; j) = 1/S(i; j)$

Local ridge frequency:

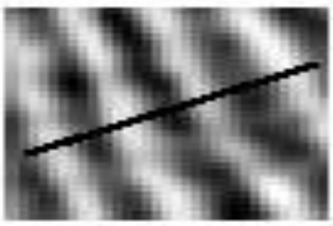


Fig. 5

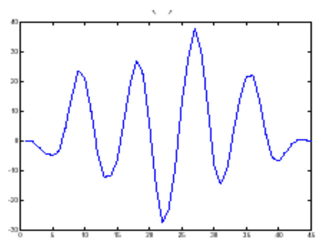


fig.6

The projection of the intensity values of the pixels along a direction orthogonal to the local ridge orientation.

(5) A 32 x 32 block from a fingerprint image.
(6) The projected waveform of the block.

5. Gabor filters:

The even-symmetric Gabor filter is the real part of the Gabor function, which is given by a cosine wave modulated by a Gaussian. An even symmetric Gabor filter in the spatial domain is defined as:

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

$$x_\theta = x\cos\theta + y\sin\theta,$$

$$y_\theta = -x\sin\theta + y\cos\theta,$$

Where θ is the orientation of the Gabor filter, f is the frequency of the cosine wave, ¾*x* and ¾*y* are the standard deviations of the Gaussian envelope along the x and y axes, respectively, and $x\theta$ and $y\theta$ define the x and y axes of the filter coordinate frame, respectively.
The Gabor filter is applied to the fingerprint image by spatially convolving the image with the filter. The convolution of a pixel ($i; j$) in the image requires the corresponding orientation value $O(i; j)$ and ridge frequency value $F(i; j)$ of that pixel. Hence, the application of the Gabor filter G to obtain the enhanced image E is performed as follows:

$$E(i,j) = \sum_{u=-\frac{w_x}{2}}^{\frac{w_x}{2}} \sum_{v=-\frac{w_y}{2}}^{\frac{w_y}{2}} G(u, v; O(i,j), F(i,j)) N(i-u, j-v),$$

Where O is the orientation image, F is the ridge frequency image, N is the normalised fingerprint image, and wx and wy are the width and height of the
Gabor filter mask, respectively.

6. Binarisation:

Most minutiae extraction algorithms operate on binary images where there are only two levels of interest: the black pixels that represent ridges, and the white pixels that represent valleys. Binarisation is the process that converts a grey level image into a binary image. This improves the contrast between the ridges and valleys in a fingerprint image, and consequently facilitates the extraction of minutiae.

7. Thinning:

The final image enhancement step typically performed prior to minutiae extraction is thinning. Thinning is a morphological operation that successively erodes away the foreground pixels until they are one pixel wide. A standard thinning algorithm is employed, which performs the thinning operation using two sub iterations. This algorithm is accessible in MATLAB via the `thin' operation under the bimorph function. Each sub iteration begins by examining the neighborhood of each pixel in the binary image, and based on a particular set of pixel-deletion criteria, it checks whether the pixel can be deleted or not.

2. Minutiae extraction:

The most commonly employed method of minutiae extraction is the Crossing
Number (CN) This method involves the use of the skeleton image where the ridge flow pattern is eight-connected. The minutiae are extracted by scanning the local neighborhood of each ridge pixel in the image using a 3£3
window. The CN value is then computed, which is defined as half the sum of the differences between pairs of adjacent pixels in the eight neighborhood. Using the properties of the CN as shown in Table 2.1

Table 2.1

CN	Property
0	Isolated point
1	Ridge ending point
2	Continuing ridge point
3	Bifurcation point

3. Verification:

At the verification stage, the template from the claimant fingerprint is compared against that of the enrollee fingerprint. This is done by comparing neighborhoods of nearby minutiae for similarity. A single neighborhood may consist of three or more nearby minutiae. Each of these is located at a certain distance and relative orientation from each other. Furthermore each minutia has its own attributes of type and minutia direction, which are also compared. If comparison indicates only small differences between the neighborhood in the enrollee fingerprint

and that in the claimant fingerprint, then these neighborhoods are said to match. This is done exhaustively for all combinations of neighborhoods and if enough similarities are found, then the fingerprints are said to match. Template matching can be visualized as graph matching, that is comparing the shapes of graphs joining fingerprint minutiae.

4. Recognition Rate

The ultimate measure of utility of a fingerprint system for a particular application is recognition rate. This can be described by two values. The false acceptance rate (FAR) is the ratio of the number of instances of pairs of different fingerprints found to (erroneously) match to the total number of match attempts. The false rejection rate (FRR) is the ratio of the number of instances of pairs of the same fingerprint are found not to match to the total number of match attempts. FAR and FRR trade off against one another. That is, a system can usually be adjusted to vary these two results for the particular application, however decreasing one increases the other and vice versa. FAR is also called, false match rate or Type II error, and FRR is also called false non-match rate or Type I error. These are expressed as values in [0, 1] interval or as percentage values. The ROC-curve plots FAR versus FRR for a system. (ROC stands for Receiver Operating Curve for historical reasons. Yes, "ROC-curve" is redundant, but this is the common usage.) ROC-curves are shown in Figure 4.1 The FAR is usually plotted on the horizontal axis as the independent variable. The FRR is plotted on the vertical axis as the dependent variable. Because of the range of FAR values, this axis is often on a logarithmic scale. Figure 2.4 contains two solid curves and three dotted curves. The solid curves do not represent any particular data; they are included for illustrative purposes to show better and worse curve placements. The typical ROC-curve has a shape whose "elbow" points toward (0,0) and whose asymptotes are the positive x and

y-axes. The sharper the elbow and (equivalently) the closer is the ROC-curve to the x- and y-axes, the lower is the recognition error and the more desirable is the result.

Figure 4.1 ROC-curves. The 2 solid curves are of hypothetical data illustrating desirable and less desirable recognition performance. The 3 dotted curves are of real data measuring the performance of 3 commercial AFIS [46].

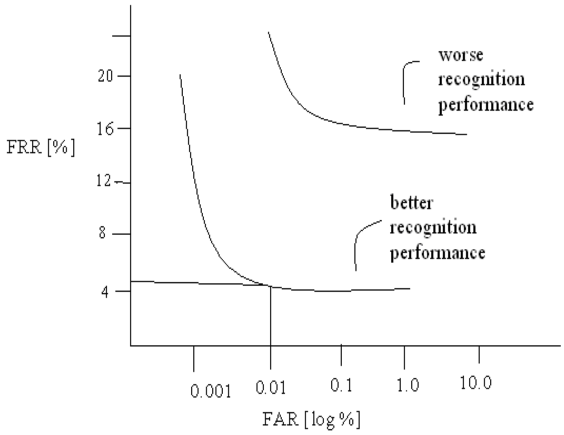


Fig 4.1

The procedure for using the ROC-curve is as follows. Choose an acceptable level of FAR. On Figure 4.1, a dashed line is shown at 0.01% FAR. The FRR corresponding to this choice is the attainable FRR, in this example about 4%. Alternatively, the FRR can be specified and the FAR found on the curve. There is no single set of FAR and FRR specifications useful for all different applications. If the fingerprint system is specified for very high security situations such as for military installations, then the FAR will be chosen to be very low (e.g., <0.001%). However, this results in higher FRR, sometimes in the range from 5% to 20%. Typical customer applications such as for automatic teller machines cannot afford to alienate users with such a high FRR. Therefore, the choice in these applications is low FRR (e.g., <0.5%), at the sacrifice of higher FAR. (An FRR specification that is sometimes quoted for automatic teller machines is less than 1 per 100,000 false rejections.)

5. Conclusion:

Fast fingerprint enhancement algorithm which can adaptively improve the clarity of ridge and furrow structures based on the local ridge orientation and ridge frequency estimated from the inputted images. The performance of the algorithm was evaluated using the goodness index of the extracted minutiae and the performance of an online fingerprint verification system which incorporates this fingerprint enhancement algorithm in its minutiae extraction module. In the matching technologies like face, voice, signature are appropriate for 1-to-1 matching, only fingerprint technology is proven to have acceptable recognition rates to be practical for 1-to-many matching. This is an indication that fingerprint matching provide the highest recognition rates for verification minutia based verification

References:

[1]. A. Bazen and S. Gerez. Systematic methods for the computation of the directional field and singular points of fingerprints.
[2]. IEEE Trans. PAMI, 24(6), June 2002.
[3]. D. Maio, D. Maltoni, R. Cappelli, J. 2.Wayman, and A. Jain. FVC2000: Fingerprint verification competition. IEEE Trans. PAMI, 24(3):402–412, Mar. 2002.
[4]. A. K. Jain and F. Farrokhnia, Unsupervised Texture
[5]. Segmentation Using Gabor Filters, Pattern Recognition,
[6]. Vol. 24, No. 12, pp. 1167-1186, 1991.
[7]. 4. A. K. Jain, L. Hong, and R. Bolle, “On-line fingerprint verification,” IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 19, No. 4, pp. 302-313, 1997.
[8]. 5. A. K. Jain, L. Hong, S. Pankanti, and R. Bolle, “An Identity-Authentication System UsingFingerprints,” Proceedings of the IEEE, Vol. 85, No. 9, pp. 1365-1388, 1997.